2020 IEEE International Conference on Robotics and Automation (ICRA)
31 May - 31 August, 2020. Paris, France

FG-GMM-based Interactive Behavior Estimation for Autonomous Driving Vehicles in Ramp Merging Control*

Yiwei Lyu[1], Chiyu Dong[2], and John M. Dolan[3]

***Abstract*— Interactive behavior is important for autonomous driving vehicles, especially for scenarios like ramp merging which require significant social interaction between autonomous driving vehicles and human-driven cars. This paper enhances our previous Probabilistic Graphical Model (PGM) merging control model for the interactive behavior of autonomous driving vehicles. To better estimate the interactive behavior for autonomous driving cars, a Factor Graph (FG) is used to describe the dependency among observations and estimate other cars' intentions. Real trajectories are used to approximate the model instead of human-designed models or cost functions. Forgetting factors and a Gaussian Mixture Model (GMM) are also applied in the intention estimation process for stabilization, interpolation and smoothness. The advantage of the factor graph is that the relationship between its nodes can be described by self-defined functions, instead of probabilistic relationships as in PGM, giving more flexibility. Continuity of GMM also provides higher accuracy than the previous discrete speed transition model. The proposed method enhances the overall performance of intention estimation, in terms of collision rate and average distance between cars after merging, which means it is safer and more efficient.**

I. INTRODUCTION

Since autonomous driving vehicles cannot replace all human driven-cars in the near future, they must share roads with human drivers. Therefore, understanding human drivers' intentions is essential for safe autonomous driving, especially in interactive scenarios. Ramp merging is one of the most important and typical scenarios where autonomous vehicles interact with human-driven cars. An autonomous vehicle may collide if it can't correctly predict human drivers' intentions given a complicated situation. Human-driven cars can introduce significant uncertainty in autonomous-human driving interactions, so it is challenging to estimate their intentions.

Current autonomous driving vehicles are not able to understand social behavior properly. Instant speed measurements via RADAR make it straightforward to obtain the kinematic information of an object, but it is hard to understand the intention of a car. In addition, the output of RADAR introduces large noise that results in serious estimation oscillation. This task is more challenging if there is no V2X communication system. In our previous research, a probabilistic graphical model (PGM) [1] is proposed to model uncertainty with probabilities. By taking advantage of the independence and conditional independence that hold among random variables, the representational and computational bottlenecks are alleviated.

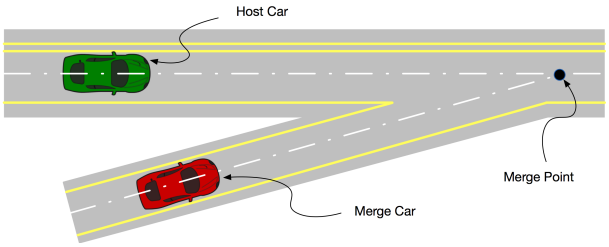

Fig. 1: Ramp merging scenario. The host car (green) is an autonomous vehicle, running on the main road; the merge car (red) is a human driven car, running on the ramp.

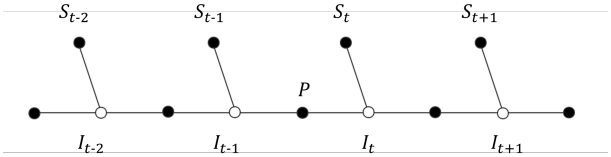

Fig. 2: A factor graph representation for the ramp merging intention estimation. S_{t-2}, S_{t-1}, S_t and S_{t+1} are factor nodes which take velocity observation and time-stamps; I_{t-2}, I_{t-1}, I_t and I_{t+1} are latent variables which denote intentions over time; P is a prior factor, which denotes the effect of prior intention.

However, one of the major drawbacks of PGM is that the relationship between nodes in PGM has to be probabilistic, i.e., all transition models should integrate to one. In addition, PGM cannot treat historical data with different attentions.

In this paper, a method that is based on Factor Graphs [2], [3] is proposed. Our main contributions are: 1) using factor graphs to better explain the relationship between variables over time; 2) using forgetting factor to pay more attention on the latest observations; 3) using Gaussian Mixture Model (GMM) to generalize the statistics which gives a smooth transition model.

II. RELATED WORK

There are several works that address the merging problem. Adaptive Cruise Control (ACC) is one approach which is being widely commercialized. For example, GM's Full Speed Range ACC and Audi's "STOP and GO" ACC make it possible for cars to follow other cars in dense traffic with low speed. ACC is also used by Mercedes-Benz in its lane departure prevention system and Tesla in "Autopilot" to perform higher-speed autonomous driving. However, these

*This work was supported by NSF REU funding.

[1]Yiwei Lyu is with the Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213 USA.

[2]Chiyu Dong is with the Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA 15213 USA.

[3]John M. Dolan is with the Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213 USA.

methods cannot help vehicles generate social behaviors to interact with other cars properly.

Slot-based approaches are proposed by Marinescu et al. [4] and Baker et al. [5], for cooperative intelligent vehicles in on-ramp or intersection traffic merging. Their contribution includes preventing congestion under heavy traffic conditions by predicting the time to arrival to the merging point accurately. However, their decision is based on current states and no historical data are considered, which can lead to failures in some cases. Nilsson et al. [6] and Liu et al. [7] formulated the cooperative planning problem as Model Predictive Control (MPC). However, a driver model is also required in the optimization-based approaches. Wei et al. [8] used a hard-coded distribution and cost function to estimate merging intention. They proposed an intention-integrated framework to enable an autonomous car to perform cooperative social behavior, but only instantaneous acceleration is considered. The lack of historical data leads to instability in estimated intention, which results in oscillation or delayed reaction to the autonomous vehicle. Galceran et al. [9] used a MDP-like framework for lane-change or merge-in. The method is theoretically sound, but may take a long time to converge if the scenario has not been solved before. Kuefler et al. [10] used Generative Adversarial Networks (GAN) mostly for highway single-lane interactive driving and distance keeping. Zhan et al. [11] used a set of prototype trajectories with probabilities to represent the prediction and safety guarantees.

However, estimating the full trajectory is not always necessary in some scenarios and also takes computational power. Deep Reinforcement Learning (DRL) is also applied to the interactive behavioral planning. Isele et al. [12] and Qiao et al. [13] mainly focus on intersection control. The DRL framework may be limited by specific setup of a scenario, e.g., the size and shape of an intersection. Dong et al. [1] proposed a PGM learning-based method for intention estimation. The main contribution is that the model is trained and validated by real trajectories, and shows great improvement when validated by a designed merging strategy in simulation compared with previous methods. However, the model only takes speed and time as factors in the process of intention estimation. In addition, the transition models trained from data still rely on the Markov assumption.

We use a factor graph which is converted from the original probabilistic graphical model (PGM) to describe dependency among observed data and estimate other car intentions. Real driving data are used to parameterize this model instead of manually designed parameters. The social behavior estimation can be extended to various cooperation situations, such as lane changing, stop sign traversal and ramp merging. In this paper, we focus on ramp merging.

III. Factor Graph-Based Intention Estimation

A. Structure of Factor Graphs

Factor graphs are widely used in perception [14] and motion planning [15]. We here adapt them to behavioral estimation. A factor graph representation for the ramp merging intention estimation is shown in Fig. 2.

Our model assumes that human intention does not oscillate as fast as the program's update rate. Therefore, one intention node will affect the next few n speed nodes. A few speed nodes can reflect an instant intention consistently. These n speed nodes keep track of the target vehicle's speed during the last n cycles. Both past states (speed) and last intention will decide the intention. We also assume that the required model inputs, including speed and distance to the host vehicle of surrounding merging vehicles, can be accurately obtained via sensors and systems.

$$P(I_{t+1}|I_t, V_{t-n}, ..., V_t) \tag{1}$$

where I_t is the last intention, and V_{t-n}, ..., V_t are the past $n+1$ velocities. Alternatively, Equation 1 can be written as,

$$I_{t+1} = \arg\max_{I_{t+1}} f(I_{t+1}, I_t, V_{t-n}, ..., V_t) \tag{2}$$

where $f(I_{t+1}, I_t, V_{t-n}, ..., V_t)$ is a factorization of current intention, last intention, and the past $n+1$ velocities.

There are two kinds of effects on future intention estimation from other nodes/edges in the factor graph.

1) State (speeds) Effect: The state effect is indicated in Equation 3, which takes historical speed data into consideration during the future intention estimation process. The forgetting function $\omega(i)$ is included in the factor nodes and assigned to historical speed data. The forgetting function is an exponential term with predefined parameters with respect to the frame timestamps of historical speed data we take into consideration. Each frame of selected historical speed data will affect the current intention estimation with different weight assigned by the forgetting function. The forgetting function makes it possible that the closer the historical speed is to the current one, the larger effect it will have on current intention estimation.

$$g(S_{t+1}) = \prod_{i=1}^{n} \omega(i) f(V_i, V_{i-1}) \tag{3}$$

where $f(V_i, V_{i-1})$ describes the probabilistic transition from the last velocity to the current one, given a certain intention hypothesis. $f(V_i, V_{i-1}) = P(V_i|V_{i-1}, I_i)$, and $g(\cdot)$ denotes the discounted influence that the state (i.e., the intention) applies to the change of speeds. S_t consists of the past n speeds alone with their timestamps.

Here $(V_n, ..., V_i, V_0)$ correspond to the last n+1 frames of past velocity $(V_t, V_{t-1}, ..., V_{t-n})$. The transitional relationship for each current-past velocity pair is gained from the Speed Transition Model, which is introduced in a previous paper [1].

2) Last Intention Effect: This is the prior intention's effect on the future intention estimation, which is indicated in Equation 4. The $b(I_t)$ and $b(I_{t+1})$ in Equation 4 are a "Blurring function", which makes the discrete intention a continuous value ranging from 0 to 1. The Blurring functions help to smooth the estimated intention and encourage the stability of intentions over time.

$$m(I_t) = \exp\{-\frac{||b(I_{t+1}) - b(I_t))||^2}{\sigma}\} \quad (4)$$

where $m(I_t)$ denotes the effect that the last estimated intention applies to the current prediction. σ is the variance of the Gaussian Distribution.

The Factorization F = $f(I_{t+1}, I_t, V_{t-n}, ..., V_t)$ is expressed as:

$$\begin{aligned} F &= g(S_{t+1}) \cdot m(I_t) \\ &= \prod_{i=1}^{n} \omega(i) f(V_i, V_{i-1}) \cdot m(I_t) \\ &= \prod_{i=1}^{n} \omega(i) \exp(-\frac{||\Delta V(I_{t+1})||^2}{\sigma}) \cdot \exp(-\frac{||\Delta b(I_{t+1})||^2}{\sigma}) \end{aligned} \quad (5)$$

where $\Delta V(I_{t+1}) \triangleq V(I_{t+1}) - V(I_t)$, and $\Delta b(I_{t+1}) \triangleq b(I_{t+1}) - b(I_t)$.

B. Evaluation of Factor Graph

For the forgetting functions in the State effect, we define exponential functions to assign different weights to historical data. This has two main advantages:

1) Prevent Underflow: With exponential forgetting functions, the original product expression is converted to a summation. In the product expression, with tiny values, it's easy to have an underflow situation, in which the absolute value of the number is too close to zero for the computer to represent it. Using exponential terms instead, every term will be added up together, allowing efficient calculation and preventing the underflow.

2) Reduce Computation: Converting the original product expression to a summation with exponential terms requires less computation (summation vs. multiplication).

C. Intention Estimation Procedure

The last 20 frames of historical speed data before the merging point $(V_t, V_{t-1}, ..., V_{t-19})$ are selected to be taken into state effect consideration. Forgetting decay functions are assigned to each frame's historical speed datum. The estimated intention I_{t+1} is affected by the 20 frames of historical data, last intention I_t, and the prior term P which assigns weight to the last intention effect.

If the merging vehicle's estimated intention is 'Not Yield', it will tend to speed up and reach the merging point before the host vehicle. Then the host vehicle activates a distance-keeping model to keep a desired safe longitudinal distance to the merging car on the ramp. On the other hand, if its estimated intention is 'Yield', the host car will ignore the merging car and accelerate to the speed limit.

IV. Training from Data

In [8], prediction of the merging car's behavior was based on hard-coded cost functions and assumptions about the probability distribution of acceleration.

We instead use the US-101 and I-80 freeway real-world dataset NGSIM to extract a model of cooperative behavior between host and merging vehicles. The dataset was obtained from overhead cameras near the US-101 and I-80's entrance ramps respectively in the Los Angeles area and Emeryville areas. Cars were filmed and tracked during morning rush hours (7:50 am to 8:35 am for US101, 4:00 am to 5:30 am for I80). The road segment consists of 5 lanes and one entrance ramp at the beginning for US-101, as shown in Fig. 3 [16]. And the road segment consists of 6 lanes and one entrance ramp at the beginning for I-80.

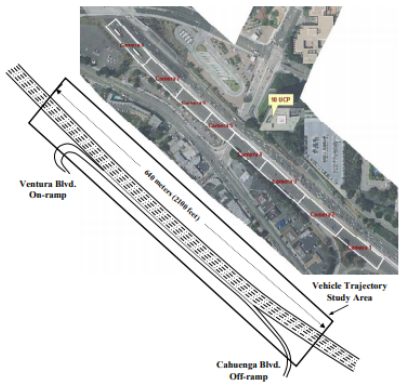

Fig. 3: The aerial photograph above shows the extent of the US 101 study area in relation to the building from which the digital video cameras were mounted and the coverage area for each of the eight cameras. The schematic drawing [16] on the bottom shows the number of lanes and location of the on-ramp at Ventura Boulevard and the off-ramp at Cahuenga Boulevard within the US 101 study area.

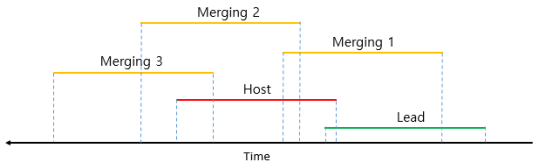

Fig. 4: Time overlap example for a host-leading pair, and multiple host-merging pairs. Three different colors, green, red and yellow indicate three classes of vehicles, the lead vehicle, the host vehicle, and merging vehicles. The length of the colored lines indicates their corresponding start time and end time for appearance in the camera.

Vehicles in the right-most lane on the main road are considered host vehicles, and counterparts on the entrance ramp are considered merging vehicles. We preprocessed the data to filter out unrelated cars that run in inner lanes without interacting with merging vehicles, and used only those from the right-most lane and the entrance ramp.

Host vehicles are paired with merging vehicles that are close to and temporally overlapped with the host, as shown in Fig. 4. There were 647 host-merging vehicle pairs in the dataset. We classify merging vehicles into two classes: 1) yield; 2) not yield, based on which car reaches the merging point first. From group 1, the distribution of $P(V|I = \mathbf{Y})$ can be estimated; from group 2, the distribution of $P(V|I = \mathbf{N})$ can be estimated.

We use 1/3 of the US-101 dataset for training, and the remaining 2/3 of the US-101 dataset and the full I-80 dataset

for real-data testing.

A. Speed Transition Model

The goal of the training is to estimate the conditional probability of intentions given historical speed information, i.e., $P(I|V)$. The two classes of data are used to train the speed transition model. Fig. 5 show the speed transition probability distributions for given speeds and intentions.

The x-axis V_{t-1} indicates the previous speed of merging vehicles, and the y-axis V_t indicates the current speed of merging vehicles. The darkness of the color zone shows the probability of a particular speed transition occurring.

The main difference between the two figures is in the zone 15 feet/second $\leq V(t-1) \leq$ 35 feet/second and 15 feet/second $\leq V(t) \leq$ 35 feet/second. The darker parts indicate the differences between these two distributions (likelihood), i.e., the change of speeds given a certain intention. In the case of "Not Yield", cars are more likely to run at higher speeds; in the case of "Yield", they are more likely to run at lower speeds. If the merging car's driver decides not to yield to the host, it has higher probability to accelerate; and the merge car is more likely to decelerate if it decides to yield to the host. These results are consistent with intuition. Additionally, each speed has specific transition probabilities under different intentions.

B. Gaussian Mixture Model (GMM)

The speed transition model is also used in the previous PGM-based method. However, there are several shortcomings of the speed transition model:

1) Inefficient Computation Process: In the speed transition model, given current and past speed, a closest coordinate is searched in the two spaces and its probability of occurrence is taken for comparison between 'Yield' or 'Not Yield'. The searching process takes time and is inefficient.

2) Missing Data: In some cases, an existing point close to the target coordinate may not easily be found. In the speed transition model, surrounding points' average probability of occurrence are calculated. This may also lead to inaccuracy. To tackle the problems above, a Gaussian Mixture Model [17] is used to approximate the speed transition probability distribution. A 2-component GMM is built, as shown in TABLE I.

We assume the distribution of speed transitions as a mixture of two single Gaussian models. Means μ, covariance σ, and weights ω gotten from the GMM are used to calculate the probability for yield and not yield:

$$P(I) = \sum_i A_i \exp \frac{||x - \mu_i||^2}{\sigma_i}, \tag{6}$$

where

$$A_i = \frac{\omega_i}{\sqrt{2\pi}\sigma_i} \tag{7}$$

V. Experimental results

We run simulations on reacting to merging vehicles with real-data trajectories which are extracted from datasets. The host car will estimate the merging vehicles' intention and make the decision by observing their state.

Vehicles on the main road and ramp have the same task: they should cooperate to merge together safely and efficiently. Therefore, regardless of the main/ramp road geometry, the ramp merging problem is topologically symmetric, so there is no difference if we make the main road or the ramp vehicle autonomous. In our experiments, we put the autonomous vehicle on the main road.

The host car uses Factor Graph and Gaussian Mixture Model (FG-GMM) for intention estimation and ACC for distance keeping, and we apply real trajectories to the merging cars.

A. Validity Test of FG-GMM Model

To prove the validity of our FG-GMM model, simulations are run on single datasets of I80 and US101.

100 host-merging pairs extracted from the I-80 dataset with time range from 05:15 am to 05:30 am and 100 host-merging pairs extracted from the US-101 dataset with time range from 07:50-08:05 were used for testing.

We conducted three sets of experiments: 1) Intention estimation without FG and GMM (PGM-G); 2) Intention estimation with FG but without GMM; 3) Intention estimation with both FG and GMM.

The experimental results for validity tests for FG-GMM are shown in TABLE II. Collision rate is calculated as the rate of total collided pair counts among all tested pairs.

From TABLE II, by comparing the first and second columns, we find that the collision rate without FG and GMM, which is the collision rate of PGM-G, is higher than the one only without GMM. The addition of FG decreases the chance for collision by an average of eight percent for the two datasets, which proves that the FG is effective. Compared with previous methods, FG uses forgetting factors to assign different weights to past velocity observations, which provides better description of past velocity information. For example, when taking past velocity information into account, a driver will not value the velocity ten seconds ago as highly as the velocity one second ago. The effect of older velocities becomes smaller, which also matches reality well.

By comparing the second and third columns, we find that the collision rate without GMM is much higher than the one with GMM. The presence of GMM decreases the chance for collision by another seven percent on both of the two datasets. Compared with the previous speed transition model, one of the features of the two-component GMM is that we assume the probability distribution for speed transition pairs is a mixture of two single Gaussian Models for the binary decision, yield or not yield. This fills the gap of possible wrong decisions caused by missing points in the previous speed transition model. These test results indicate that the GMM greatly improves the accuracy of estimated intention.

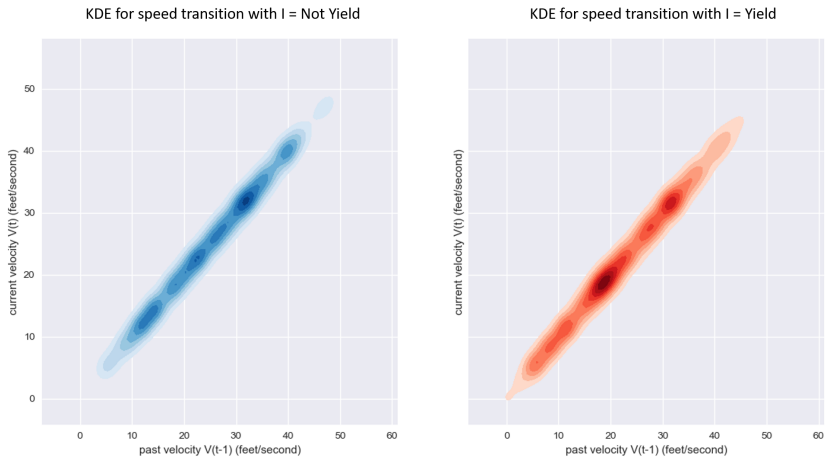

Fig. 5: Speed transition distribution for given speed and 'Yield' / 'Not Yield' intention. Notice the different locations where high probability yields in two situations.

TABLE I: 2-component Gaussian Mixture Model parameters w.r.t. different intentions.

Yield		Not Yield	
Means	[[17.59 17.58][43.51 43.50]]	Means	[[44.02 44.01][17.87 17.87]]
Covariance	[[64.04 64.00][106.53 106.77]]	Covariance	[[106.96 107.20][66.25 66.20]]
Weights	[0.57 0.42]	Weights	[0.43 0.56]

TABLE II: Collision Rate Comparison

Dataset	Num. of Pairs	Collision Rate(w/o FG, GMM)	Collision Rate(w/o GMM)	Collision Rate(w/ GMM)
I-80 (05:15-05:30)	100.0	23.0	14.0	7.0
US-101 (07:50-08:05)	100.0	20.0	13.0	6.0

In conclusion, the FG-GMM model we propose achieves improved performance in intention estimation for self-driving cars.

B. Comparison of FG-GMM and previous methods

Tests are run on the remaining 2/3 of the US101 dataset and the full I80 dataset. We compare our new algorithm with the following methods:

1) *ACC merging*, a non-cooperative method that distance-keeps to the merging car if it is closer to the merging point;
2) *Slot checking*, which is adopted from the Urban Challenge [4].
3) *iPCB*, which is proposed in [8].
4) *PGM-G*, which uses the proposed PGM structure, but assumes a Gaussian Distribution for the speed transition probability, like iPCB.

We use two criteria to verify the performance of these algorithms: 1) failure rate based on number of collision scenarios; 2) average minimum distance between the host and nearest merging car when the host reaches the merging point. The first criterion deals with safety, and the second one deals with efficiency. The comparison result is shown in TABLE III.

The collision rates are shown in TABLE III in the first halves of the "US-101 Data" and the "I-80 Data" columns.

TABLE III: Statistical results for different methods

Method	US-101 Data		I-80 Data		Cycle rate
	%	D(m)	%	D(m)	ms
ACC	17.6	22.2	16.5	7.0	0.05
Slot	14.8	22.8	10.4	10.3	N/A
iPCB	19.3	23.7	15.8	13.7	0.20
PGM-G	20.1	24.3	11.3	14.6	0.51
PGM	8.7	25.8	7.6	16.4	0.08
FG-GMM	7.3	23.3	7.0	14.7	0.08

FG-GMM's collision rate was 1.4% lower than PGM on the US-101 dataset and 0.6% lower on I80. Therefore, FG-GMM with the lowest collision rate is the safest one among all the methods. The D(m) columns in the second halves of the "US-101 Data" and the "I-80 Data" columns show average distance between the host cars and the closest merging cars when the host car reaches the merging point. FG-GMM's average distance is 9.6% shorter than PGM on US-101 and 10.3% shorter on I-80. Comparing with PGM, FG-GMM has shorter average distance in both datasets, which means that it has the higher degree of traffic efficiency. Though its average distance is higher than ACC, Slot, iPCB, and PGM-G, considering the collision rates of these methods are more than two times of FG-GMM, FG-GMM still has a good overall safe performance.

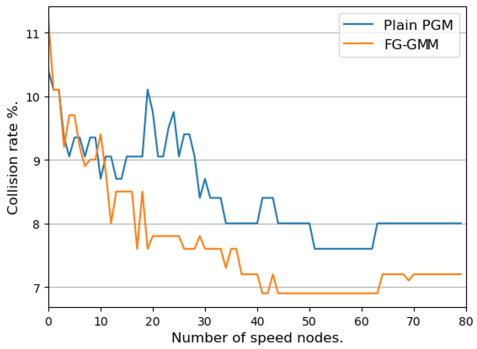

Fig. 6: Collision rates v.s. different number of speed nodes.

In conclusion, FG-GMM achieves better experimental results than previous methods, in terms of collision rate, average distance and overall performance. Since I-80 has a longer merging ramp than US-101, the host vehicle has more time to interact with the merging vehicle, and the controller also has enough time to adjust the vehicle to a certain speed. This may explain why the experimental results on I-80 are slightly better than those on US-101.

C. Collision Rates vs. Number of Speed Nodes

In order to determine the proper number of speed nodes, the proposed method was applied to the dataset, with a varying number of speed nodes. Fig. 6 shows the collision rate as we increase the number of speed nodes. For FG-GMM, the collision rate firstly decreases from 1 to 45, keeping the same from 45 to 64, and then slightly go up after 64. It is no surprise that more speed information helps the decision making. However, excess past information can reduce sensitivity to the present dynamic changes, which is the reason why the collision rates increase as we use more than 64 nodes. Thus, choosing a proper number of speed nodes is a trade-off between robustness, sensitivity and computational efficiency. Therefore, 45 speed nodes are preferred to 64. Note that each node captures one speed measurement whose update rate is 10Hz. Thus 45 nodes require 4.5s past measurements for estimates. Comparing to the plain PGM plot of Collision rate vs. number of speed nodes in previous work [1], the collision rate keeps decreasing instead of significantly going up before reaching the optimized nodes number, and the optimized range also moves towards zero, which means it approaches the convergence in a faster speed. This is the result of using forgetting factors in FG-GMM.

VI. Conclusion

Real data test results show that the proposed FG-GMM method has the lowest collision rate comparing to previous methods, including ACC, Slot, i-PCB, and PGM, FG-GMM. It has the best overall performance, therefore, Factor Graph is safer and more efficient.

In the future, we will test the generality of our algorithm against a different merging behavior through a manually designed merging strategy. Currently, the behavior prediction is not robust enough, and the planning algorithm cannot work without the prediction under uncertainty. Future goals are to enhance the accuracy of the prediction and to develop a framework that can combine the planning and prediction into a closed-loop system for future cooperative autonomous planning. We will also extend our method to estimate long-term motion of merging vehicles, in scenarios that have more than one merging car, and take advantage of a broader set of training data.

Acknowledgments

The authors would like to thank Junqing Wei and Wenda Xu, who contributed to previous ideas of ramp merging.

References

[1] C. Dong, J. M. Dolan, and B. Litkouhi, "Intention estimation for ramp merging control in autonomous driving," in *2017 IEEE 28th Intelligent Vehicles Symposium (IV'17)*, Jun. 2017, pp. 1584 – 1589.

[2] H.-A. Loeliger, "An introduction to factor graphs," *IEEE Signal Processing Magazine*, vol. 21, no. 1, pp. 28–41, 2004.

[3] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Transactions on information theory*, vol. 47, no. 2, pp. 498–519, 2001.

[4] D. Marinescu, J. Čurn, M. Bouroche, and V. Cahill, "On-ramp traffic merging using cooperative intelligent vehicles: A slot-based approach," in *Intelligent Transportation Systems (ITSC), 2012 15th International IEEE Conference on*. IEEE, 2012, pp. 900–906.

[5] C. R. Baker and J. M. Dolan, "Traffic interaction in the urban challenge: Putting boss on its best behavior," in *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*. IEEE, 2008, pp. 1752–1758.

[6] J. Nilsson, M. Brannstrom, E. Coelingh, and J. Fredriksson, "Lane Change Maneuvers for Automated Vehicles," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–10, 2016.

[7] C. Liu, C.-Y. Lin, Y. Wang, and M. Tomizuka, "Convex feasible set algorithm for constrained trajectory smoothing," in *American Control Conference (ACC), 2017*. IEEE, 2017, pp. 4177–4182.

[8] J. Wei, J. Dolan, and B. Litkouhi, "A prediction-and cost function-based algorithm for robust autonomous freeway driving," in *Intelligent Vehicles Symposium*, 2010.

[9] E. Galceran, A. G. Cunningham, R. M. Eustice, and E. Olson, "Multipolicy decision-making for autonomous driving via changepoint-based behavior prediction: Theory and experiment," *Autonomous Robots*, 2017, in Press.

[10] A. Kuefler, J. Morton, T. Wheeler, and M. Kochenderfer, "Imitating driver behavior with generative adversarial networks," in *Intelligent Vehicles Symposium (IV), 2017 IEEE*. IEEE, 2017, pp. 204–211.

[11] W. Zhan, L. Sun, Y. Hu, J. Li, and M. Tomizuka, "Towards a fatality-aware benchmark of probabilistic reaction prediction in highly interactive driving scenarios," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2018, pp. 3274–3280.

[12] D. Isele, A. Cosgun, K. Subramanian, and K. Fujimura, "Navigating intersections with autonomous vehicles using deep reinforcement learning," *CoRR*, vol. abs/1705.01196, 2017. [Online]. Available: http://arxiv.org/abs/1705.01196

[13] Z. Qiao, K. Muelling, J. M. Dolan, P. Palanisamy, and P. Mudalige, "Automatically generated curriculum based reinforcement learning for autonomous vehicles in urban environment," in *Intelligent Vehicles Symposium (IV), 2018 IEEE*. IEEE, 2018, pp. 1233–1238.

[14] F. Dellaert, M. Kaess *et al.*, "Factor graphs for robot perception," *Foundations and Trends® in Robotics*, vol. 6, no. 1-2, pp. 1–139, 2017.

[15] J. Dong, M. Mukadam, F. Dellaert, and B. Boots, "Motion planning as probabilistic inference using gaussian processes and factor graphs." in *Robotics: Science and Systems*, vol. 12, 2016.

[16] V. Alexiadis, J. Colyar, J. Halkias, R. Hranac, and G. McHale, "The next generation simulation program," *Institute of Transportation Engineers. ITE Journal*, vol. 74, no. 8, p. 22, 2004.

[17] N. M. Nasrabadi, "Pattern recognition and machine learning," *Journal of electronic imaging*, vol. 16, no. 4, p. 049901, 2007.