A Probabilistic Framework for Microscopic Traffic Propagation

Tim A. Wheeler
Aeronautics and Astronautics
Stanford University
Stanford, California 94305
wheelt@stanford.edu

Philipp Robbel
Robert Bosch LLC
Palo Alto, California 94304
philipp.robbel@us.bosch.com

Mykel J. Kochenderfer
Aeronautics and Astronautics
Stanford University
Stanford, California 94305
mykel@stanford.edu

Abstract—Probabilistic microscopic traffic models provide a statistical representation of interactive behavior between traffic participants. They are crucial for the validation of automotive safety systems that make decisions based on surrounding traffic. The construction of such models by hand is error-prone and difficult to extend to the complete diversity of human behavior. This paper describes a methodology for microscopic traffic model construction based on a Bayesian statistical framework connected to real-world data and applies it to learning models for free-flow, car following, and lane-change behaviors on highways. The evolution of traffic scenes is represented by a generative model learned for individual vehicles that captures their response to other traffic participants as well as the road structure. Our evaluation shows realistic behaviors over a four second horizon. A complete implementation is available online.

I. INTRODUCTION

Modern road vehicles employ a variety of safety systems including lane departure warning, forward collision avoidance and mitigation, blind spot monitoring, and collision imminent braking systems. As the automotive industry moves towards autonomous driving, these safety systems will act with increasing autonomy and engage in increasingly complicated driving interactions. Confidence in the performance and robustness of these safety systems is required before public deployment due to the potentially fatal consequences of error in their operation.

Recently, models of aircraft behavior have been developed to validate collision avoidance systems [1]. These models used dynamic Bayesian networks to represent encounters between aircraft. Model structure and parameters were learned from a large corpus of radar data. Use of these models in optimal policy generation resulted in significant reductions in collision rates and pilot alert rates [2]. This paper adapts this approach from aviation and uses it in the derivation of probabilistic driving models.

Use of probabilistic action models in microscopic traffic simulation has also been the subject of previous research. Acceleration models for car following have been studied extensively since the 1950s [3]. Specific models have been developed for many scenarios, some of which include lane-changing [4], merges [5], and emergency braking [6]. These models assume specific features and forms for the response equations, and lateral control is often limited to a binary lane change decision.

Recent work has outlined a high level framework for probabilistic driving models and seeks to automate the construction of microscopic action models from data. In [7], a softmax classifier based on a linear feature weighting is used to identify a context class tied to a Gaussian acceleration distribution. In [8], a microscopic action model is constructed using random forests. Their work modeled a Gaussian distribution over acceleration and turn-rate conditioned on local context. The resulting distribution is necessarily unimodal and was only tested on a simulated scenario with two vehicles.

The probabilistic models developed in this paper exhibit several important properties. First, these models are automatically learned from naturalistic driving data from various traffic settings. Second, unlike maneuver-recognition approaches, these models capture microscopic acceleration and turn-rate behavior on a per-vehicle basis. Third, behavior is modeled using a Markov model represented as a dynamic Bayesian network, which can fit arbitrary distributions with sufficient discretization. Feature selection is a natural result of structure search, allowing for models to be automatically learned for specific driving scenarios or geographic regions.

II. PROPAGATION MODEL OVERVIEW

Let a scene $s_t$ define the joint configuration of vehicles at a particular time $t$ and any past information necessary to satisfy the Markov assumption. A propagation model represents the distribution $P(s_{t+1} | s_t)$, relating a scene to the distribution over successor scenes. The objective of this paper is to establish a framework for obtaining an accurate distribution over future traffic scenes from real-world driving data.

Varying roadway composition and traffic participant quantities make directly modeling a distribution over future scenes difficult. Recent approaches develop microscopic action models applied on a per-vehicle basis based on local context extracted from the current scene, $P(a_i^{(t)} | s_t, b_i^{(t)})$, where $a_i^{(t)}$ is the action taken by the $i$th vehicle and $b_i^{(t)}$ is its identified behavior [5], [8]. Probabilistically valid future scenes can be obtained by sampling from the action distribution for each traffic participant and propagating each vehicle over a small time-step using a dynamics model.

Prior work on aircraft propagation models used dynamic Bayesian networks (DBNs) to represent the action distribution
An ideal microscopic traffic model produces a realistic action distribution for any possible traffic scenario. Accomplishing this with a single model is difficult, as different vehicle behaviors are best described by different indicators. Different models have traditionally been developed for different behaviors [3]–[6]. Our method follows this approach and yields a general framework that can learn suitable structures and parameters for any behavior class covered by the data set. In this work, we restrict our focus to three scenarios that are particularly important for driving applications: free-flow, the vehicle may drive unimpeded in its lane at its desired speed; car following, the vehicle follows its lane while keeping pace with another vehicle; and lane-changing, the vehicle selects and moves into another lane. Vehicle behaviors are identified at each time step so the appropriate model can be applied. Behavior class definitions are given in Table I. Free-flow and car following are mutually exclusive whereas lane-changes occur during free-flow or car following.

### III. DATA SOURCE AND FEATURES

This work uses real-world highway driving data collected in the San Francisco Bay Area of Northern California and Detroit, Michigan. Approximately two hours of post-processed driving data were available for model construction.

Drives were conducted with a sensor-equipped passenger vehicle allowing precise ego motion estimation and tracking of surrounding vehicles through 360 degree LIDAR and radar sensor coverage. Data with motion estimates for ego and surrounding vehicles sampled at 20Hz were made available for this work. Position and velocity traces were recorded for the ego-vehicle in global coordinates and for surrounding traffic participants in an ego-relative frame. Training data was restricted to that of the ego-vehicle as the surroundings of other vehicles are often occluded, making reasoning about their actions difficult. Future models should learn from a variety of vehicles to capture behavior differences. Data processing details are available at the link listed at the end of the paper.

#### A. Feature Extraction

A set of 143 candidate features were extracted for the ego-vehicle from lane-relative tracks. The set includes core features from ego dynamics, roadway features, relative features between vehicles, past states, and aggregate features over a vehicle’s history. Core features describe the current state of the ego vehicle, including such features as velocity and turn-rate. Roadway features are measured with respect to the nearest center lane and require knowledge of the surrounding roadway structure. Relative features between vehicles include time and distance headways and other relative measurements required for interactive traffic prediction. Features dependent on past actions are included as well, and require recording these values when propagating a vehicle in simulation. Aggregate features include the standard deviations, minimum, and maximum values for acceleration and turn-rate over various time histories, and require more detailed traces to be recorded.

This set of candidate features reflects those used in the driving literature for intention estimation [8], [11], [12]. For a complete list of candidate features and their definitions see the supplemental online material.
Continuous features were discretized using bin boundaries chosen using the variables’ marginal distributions. Discretization boundaries are available in the supplemental material.

B. Target Variables

The target variables in the microscopic model are the acceleration ($a_{\text{fut}}$) and turn-rate ($\psi_{\text{fut}}$) over the next quarter-second time step, corresponding to the traditional throttle/break and steering wheel driving inputs. The mean acceleration over the following quarter-second horizon was extracted using the velocity difference, $a_{\text{fut}} = (v_{t+\Delta T} - v_{t})/\Delta T$.

Models trained using a direct turn-rate output exhibited high sensitivity when traveling at highway speeds, as slight perturbations could lead to large lateral deviations. Improved performance was gained by propagating vehicles using a turn-rate computed from a desired lane-relative heading, $\psi_{\text{fut}} = \psi_{\text{des}} - \psi$, where a desired lane-relative heading $\psi_{\text{des}}$ of zero causes the vehicle to drive parallel to the closest centerline. The final target variable set consists of the future acceleration $a_{\text{fut}}$ and the constant desired lane-relative heading over the next time-step $\psi_{\text{des}}$, extracted by solving the differential equation of motion assuming a constant turn-rate:

$$\psi_{\text{des}} = \frac{\psi_{t+\Delta T} - \psi_{t} e^{-\Delta T}}{1 - e^{-\Delta T}}. \quad (1)$$

IV. MODEL LEARNING

A representation for the probability distribution for model variables within the DBN must be chosen. Naturally discrete variables are modeled using multinomial distributions. Continuous and hybrid variables are discretized with multinomial distributions over discrete bins and uniform distributions within bins. Resulting models can match arbitrarily complex distributions with a sufficient number of bins, as has been successfully applied in aircraft encounter models [1].

A. Structure Search

Obtaining the vehicle action model requires feature identification, obtaining the DBN graph structure that best matches the observed data, and obtaining the sufficient statistics for all conditional probability tables. Statistical techniques developed since the mid-1990s in the machine learning community infer model structure from data [13]. Structure search for a Bayesian network seeks the graph structure $G$ given the dataset $D$ that maximizes $P(G \mid D)$ [14]. We make the common assumption of a uniform prior over structures such that $P(G)$ is constant, and use the K2 parameter prior which assigns a baseline uniform distribution over conditional probability table statistics [10]. For numerical reasons it is more efficient to maximize the log-likelihood $\ln P(G \mid D)$, also known as the Bayesian score [15]. The Bayesian score allows model complexity to scale with the amount of training data. This is especially important in feature selection, as the amount of data will affect how many features are selected.

B. Efficient Feature Selection

Given infinite data and computing resources, one could conceivably use the entire set of 143 candidate features in the DBN. Conditional probability tables grow exponentially with the number of parent variables, making this intractable for reasonable dataset sizes and computation times. We instead propose automated feature selection through structure search to optimally limit the number of features while finding the Bayesian network graph structure which best fits the data [15]. Once the graph is identified, unnecessary indicator features are discarded and the conditional probability tables for the resulting variables are populated using the posterior distribution.

The general graph search problem is NP-complete, and the space of directed acyclic graphs is superexponential in the number of variables [16]. The search space for this problem can be drastically reduced by leveraging a desired model structure. In our setting, all indicator features are observed and thus need not have parents. The only edges in the network go from indicator variables to target variables or between target variables. The Bayesian score decomposes over variable nodes [14]. Structure search can thus identify the set of parents for each target variable independently, subject to the acyclicity constraint that $a_{\text{fut}} \leftrightarrow \psi_{\text{des}}$ is not bidirectional.

Three heuristic search methods were used to identify candidate graph structures maximizing the Bayesian score: Forward Search is a baseline greedy hill-climbing procedure which
begins with an empty graph and successively adds the next edge yielding the greatest increase in Bayesian score until a local maximum is achieved. Graph Traversal augments Forward Search with edge removal (and edge reversal for $a^\text{fut} \leftrightarrow \psi_{\text{des}}$). Graph Traversal with Random Initialization increases the chance of finding better local optima by running Graph Traversal with multiple randomly initialized graph structures and selecting the best result.

Each algorithm was run on two feature sets for free-flow: the full feature set of 143 indicator features and a reduced set of 72 features, lacking the aggregate features. Random initialization was run with 100 randomly chosen networks, each with two randomly selected parents per target variable.

Forward Search and Graph Traversal produced the same graph structures. The inclusion of edge removal and reversal operations in Graph Traversal are insignificant when the graph structure is restricted to a two-layer inverted tree, but is necessary when the structure is initialized with random edges. Random Initialization found structures with the highest Bayesian scores and was thus used for all subsequent experiments.

C. Structure Learning Results

The resulting model structures for each scenario and feature set are shown in Table II. Looking at the selected indicator features, the future acceleration $a^\text{fut}$ is dependent on the current acceleration and how long the vehicle has been continuously accelerating or decelerating. The desired lane-relative heading angle $\psi_{\text{des}}$ is dependent on the current turn-rate and other lateral features such as the lateral velocity and lateral acceleration. Some features are only present in certain behavior models. For example, the time to collision with the leading vehicle, $\text{ttc}_{x,\text{front}}$, is present in the car following model but is irrelevant in free-flow.

The models learned using the larger feature set were similar to those learned using the smaller feature set. The additional aggregate features are seen in some of the models but do not significantly affect the core feature set. The number of indicator features selected in the resulting models is heavily dependent on the number of training samples. The reduced training sample count for the lane change model (~8% that of free-flow) is reflected in its smaller feature count. More detailed models can be obtained by increasing training data quantity and state-space coverage.

V. Model Application

The resulting microscopic action models can now be applied to the propagation of traffic scenes by applying the microscopic action model to each vehicle in the scene. First, a vehicle’s state and behavior are identified. Free-flow and car following can be determined from their definitions (Section III), whereas lane changing behavior can be chosen according to gap acceptance models or lateral velocity [4]. The relevant indicators are extracted and discretized. Discrete values for the target variables $a^\text{fut}$ and $\psi_{\text{des}}$ are obtained by sampling from their conditional probability distributions. Continuous values for $a^\text{fut}$ and $\psi_{\text{des}}$ can be obtained by sampling from the uniform distribution defined by the bin boundaries. Zero-binning, which involves returning zero if zero is in the bin’s domain, can reduce noise when a traffic participant is holding a steady speed or lane offset.

Two methods of control input smoothing were investigated to reduce jitter: a simple moving average, $\text{SMA}(n)$, in which the mean of the previous $n$ actions is applied, and a weighted moving average, $\text{WMA}(n)$, in which the previous $n$ actions have linearly decreasing weight. Smoothing histories up to twelve model time-steps were considered ($\Delta T = \frac{1}{8}$s). Parameters were chosen separately for $a^\text{fut}$ and $\psi_{\text{des}}$.

The optimal sampling and smoothing methods for a particular model were tuned to match the probability density over final positions obtained from simulation with the one observed in the real-world data by minimizing the Kullback-Leibler divergence between both distributions. These distributions are approximated by binning the final positions after 4s propagation into a 2D histogram based on displacement in the Frenet frame. The resulting bin counts form a discrete Dirichlet probability distribution. Cyclic coordinate descent was used to iteratively converge towards a local optimum [17].

VI. Evaluation

Traffic propagation models must be representative of the form of driving they were intended to model. Models developed for each behavior class were compared to real-world data using emergent variables from simulation.

A. Free-Flow

Actual driving segments 4s in duration were compared to trajectories propagated in simulation from the same initial conditions. The emergent variables centerline offset and speed
TABLE III: Free-flow model validation comparison results

<table>
<thead>
<tr>
<th></th>
<th>Small Model</th>
<th>Large Model</th>
<th>Real-World</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a_{\text{int}}$ sampling</td>
<td>uniform</td>
<td>zero bin</td>
<td></td>
</tr>
<tr>
<td>$a_{\text{int}}$ smoothing</td>
<td>SMA(11)</td>
<td>SMA(9)</td>
<td></td>
</tr>
<tr>
<td>$\psi_{\text{des}}$ sampling</td>
<td>uniform</td>
<td>zero bin</td>
<td></td>
</tr>
<tr>
<td>$\psi_{\text{des}}$ smoothing</td>
<td>WMA(2)</td>
<td>SMA(10)</td>
<td>WMA(3)</td>
</tr>
<tr>
<td>mean lane offset</td>
<td>0.145 ± 0.099</td>
<td>0.146 ± 0.097</td>
<td>0.141 ± 0.098</td>
</tr>
<tr>
<td>mean speed</td>
<td>29.04 ± 0.24</td>
<td>29.04 ± 0.27</td>
<td>29.03 ± 0.22</td>
</tr>
</tbody>
</table>

![Fig. 3: Qualitative comparison of tracks from free-flow models; 4s horizon, log intensity scale, 10,857 tracks each](image)

From the qualitative density comparison, it follows that the small model is a better fit to the real-world data. The density from the large model has increased variance in longitudinal displacement. The quantitative variables from both models show great similarity to one another. Both models exhibit a mean lane centerline offset approximately half a meter off from the true mean centerline offset. The mean speed value shows high variance due to the high variance in initial conditions used in the comparison. The mean timegap from the model slightly underestimates the true mean timegap.

C. Lane-Change

Lane change models were trained using 53 real-world lane changes. This small sample size produces maximum likelihood structures with low indicator feature counts. These models lack lane-relative features and thus lack the context needed to produce plausible trajectories from all initial conditions.

For demonstration purposes, a lane change model was trained by augmenting the maximum-likelihood structure for $\psi_{\text{des}}$ with additional variables known to capture road structure. The added variables include the centerline offset, the time to crossing the left lane marker, the time since crossing a lane boundary, and the quarter-second past lateral velocity. The resulting feature set is shown in Table V.

![Fig. 4: Qualitative comparison of tracks from car following models; 4s horizon, log intensity scale, 1585 tracks each](image)

Increasing the feature count results in exponential growth in the sufficient statistics required to specify the conditional probability tables. The influence of the uniform K2 prior is significantly increased in sparse models. Training counts were quadrupled to reduce these effects in this demonstration.
listened in Table VI. Figure 5 shows one hundred lane-change trajectories using the augmented lane-change model.

As expected, results using the original model without road structure features do not respect the lane boundaries and yield high centerline offsets and trajectories with continued motion towards the left. The augmented model produces plausible trajectories which follow the lane centerline and have significantly decreased positional and velocity variance. Results will directly benefit from higher data availability.

VII. CONCLUSION

This paper introduces an analytic framework for learning microscopic driving models from real-world data. Dynamic Bayesian networks were used to represent a distribution over vehicles’ lateral and longitudinal control inputs, which when applied in aggregate can be used to propagate traffic scenes. Model parameters and structure are directly inferred from data. Models for free-flow, car following, and lane change were learned from recorded data and their feasibility was assessed both qualitatively and quantitatively. All models exhibit realistic motion when evaluated over a 4s horizon. Models will be revised as more data becomes available.

Future work should investigate incorporating behavior and driver intention as latent variables and learn their descriptions from data, as has been proposed and attempted by others [8], [11], [12]. Including vehicle size characteristics would allow the model to distinguish between vehicle classes. Generative models can be used in simulation to validate or optimize candidate safety systems as was done in civil aviation [2].

As active safety systems become more prevalent, they will increasingly rely on accurate predictions of the behavior of other traffic participants. The data-driven model learning framework outlined in this paper can directly form the basis for automatically constructing such models. Complete implementation details are available at https://github.com/sisl/TrafficPropagationModel.

ACKNOWLEDGMENT

This work was sponsored by Robert Bosch LLC. Opinions, interpretations, conclusions, and recommendations are those of the authors and are not necessarily endorsed by Bosch. The authors appreciate the support and assistance provided by the Bosch Automated Driving team and gratefully acknowledge the helpful comments from anonymous reviewers.

REFERENCES