TRANSPORTMETRICA A: TRANSPORT SCIENCE
2022, VOL. 18, NO. 3, 345–366
https://doi.org/10.1080/23249935.2020.1870178

Microscopic modeling of cyclists on off-street paths: a stochastic imitation learning approach

Hossameldin Mohammed, Tarek Sayed and Alexander Bigazzi

Dept. of Civil Engineering, University of British Columbia, Vancouver, Canada

ABSTRACT

Accurate modeling of bicycles in microsimulation tools is challenging due to the limited availability of detailed data, complexity of cyclist decision-making, and heterogeneity in cycling behavior. This paper proposes an agent-based bicycle simulation method in which generative adversarial imitation learning (GAIL) is used to infer the uncertain intentions and heterogeneous preferences of cyclists from observational data. The model is tested on video-derived data of cyclists on a unidirectional path in Vancouver, Canada. In cross-validation, multivariate distributions of movement variables such as speed, direction, and spacing are similar between observed and simulated cyclist trajectories. The model also performs well in comparison to two other cyclist simulation models from the literature. The proposed approach to agent-based microsimulation is a significant advancement, with continuous, non-linear, and stochastic representation of cyclist states, decisions, and actions. The enhanced consideration of cyclist diversity is necessary for developing bicycle networks for all ages and abilities of riders.

ARTICLE HISTORY
Received 4 May 2020
Accepted 23 December 2020

KEYWORDS
Microsimulation; bicycles; agent-based model; machine learning

Introduction

Bicycle simulation models have a range of practical applications for analyzing bicycle infrastructure and operations, cyclist power and energy, and traffic safety (Heinen, Bert, and Kees 2010). However, bicycle simulation models are still underdeveloped due to a lack of understanding of how cyclists make microscopic decisions about bicycle control and interactions with other road users. To overcome this limitation, a promising approach for the microsimulation of bicycle traffic is building intelligent cyclist agents that learn hidden preferences and intentions from observational data.

The objective of this research is to develop an imitation learning model that can predict accurate bicycle trajectories for microscopic simulation by learning the heterogeneous hidden utility of cyclist agents through training on real-world demonstrations of cycling

CONTACT Alexander Bigazzi alex.bigazzi@ubc.ca Dept. of Civil Engineering, University of British Columbia, 2029 – 6250 Applied Science Lane, Vancouver, BC V6T1Z4, Canada

behavior. A key novel contribution of this paper is the investigation of continuous, nonlinear, and stochastic policy estimation processes for cyclist agents using observational data.

Background and literature review

Traffic microsimulation model development to date has focused on motor vehicle traffic. These existing tools can help guide the development of bicycle microsimulation, though the operation of bicycles differs from that of motor vehicles in fundamental ways. Bicycle traffic is more maneuverable, and more affected by operator attributes including physical capability, gender, age, and fatigue (Twaddle, Schendzielorz, and Fakler 2014). The two basic models used for motor vehicle microsimulation are car-following and lane change. These two models describe the decisions of drivers to take different guidance actions (e.g. accelerate, decelerate, change lane) based on stimuli from the environment and interacting motor vehicles (Aghabayk, Sarvi, and William 2015). The adaptation of these methods to bicycle traffic must account for differences such as the absence of lanes, which allows for flexible utilization of lateral road space by cyclists and changing lateral positions to allow for more unconstrained riding or overtaking. Most car-following models assume a stimuli response mechanism, where motor vehicle actions are highly determined by leading vehicle actions. We cannot assume these same stimuli-response models apply to bicycle traffic given the differences in independence of movement. Also, stochasticity in behavior is more pronounced in bicycle traffic, in the form of inter-cyclist (differences between cyclists due to different characteristics) and intra-cyclist (variability in behavior of an individual cyclist when encountering similar situation in different time periods) heterogeneity (Taylor and Mahmassani 1998).

The most commonly used method for modeling microscopic cyclist behavior is cellular automata (CA) (Wolfram 1983). CA uses a discrete space formulation wherein cells occupied by cyclists change according to the movement of cyclists following a set of rules controlling their motion and interactions. Jiang, Jia, and Wu (2004) developed a stochastic CA model that discretizes the environment into cells containing multiple bicycles. Jia et al. (2007) developed a CA model that differentiates fast-moving and slow-moving cyclists, and compared deterministic and stochastic rules. Gould and Karner (2009) extended the CA modeling approach for multiple lanes and multiple types of cyclists. Xue et al. (2017) developed a CA model in which each cell can represent one bicycle, and each bicycle's speed is affected by a leading bicycle's speed in preceding time steps. Tang et al. (2018) formulated a CA model that can account for group behavior. Although CA has been extensively used in the bicycle microsimulation literature, it has limitations including the lack of a continuous state-space representation. Also, CA models only allow a preset number of agent groups, each with its own set of rules, which limits the ability to model multiple levels of heterogeneity and environment dynamics (Wolfram 1985). These limitations of CA models restrict the flexibility required to model bicycle traffic heterogeneity.

Other methods of modeling microscopic cyclist operations are derived from homogeneous rules of behavior. Liang, Baohua, and Qi (2012) developed a psychological-physical force model that assumes cyclists make guidance decisions (acceleration and direction) based on forces of collision avoidance and friction between interacting cyclists. Zhao and

Zhang (2017) developed a following model that can be applied to motor vehicles, bicycles, and pedestrians by assuming the same stimuli-response behavioral mechanism, but with different parameter values. The parameters were estimated from experimental data of motor vehicles, bicycles and pedestrians moving in a circle. These methods model the average behavior of cyclists, but cannot represent inter-cyclist and intra-cyclist behavior heterogeneity.

Agent-Based Modeling (ABM) is a potentially powerful approach for realistic modeling of human-directed movements and interactions in roadway environments. ABM offers flexibility in developing models that are robust and scalable, which can capture the complexity of real-world behavior that emerges from the microscale into the macroscale (Jennings 2000). Using ABM approach allows models to capture the variability in decision-making rules in varying situations and between heterogeneous agents. For example, interacting cyclists could encounter different behavioral schemes depending on their relative values of motion variables (Mohammed, Bigazzi, & Sayed, 2019). A key task in developing ABM is identification of agent goals and strategies. One approach to this task is using a structured analytical model with parameters that govern agent behavior in different situations, selected using heuristic techniques (Baster et al. 2013; Hussein and Sayed 2017). This approach to ABM has been used in various transportation applications, including modeling the effect of 'Mobility as a Service' trends (Djavadian and Joseph 2017), solving transit network design problems (Liu and Zhou 2016), and modeling the interactions between autonomous vehicles using a multi-agent framework (de Oliveira 2017). The drawback of this approach for bicycle microsimulation is the use of pre-set rules; agents do not learn from the environment or other agents, nor do they evolve through the learning process (Abdou, Hamill, and Gilbert 2012).

Another approach to ABM development is designing intelligent adaptive agents that can learn from experience by imitating expert demonstrations and evolving their goals and strategies over time (Plekhanova 2003). To apply this approach to bicycle microsimulation, the navigation and guidance behavior of a cyclist moving along a path can be framed as a finite-state Markov Decision Process (MDP). In this framework, a cyclist is an agent interacting with the environment (roadway features, other road users, traffic controls, etc.) and executing a sequential decision process. At each time step, cyclists base their decisions on a function which is called a policy throughout this paper. Policy is defined as a mapping function from the agent's current state to its consequent action. The main component of a policy is its reward function that represents the attractiveness of potential future states to the agent, following the formal approach of (forward) reinforcement learning. Solving a reinforcement learning problem requires a defined reward function, which is usually defined by the analyst for *ad hoc* control tasks. For the problem of modeling cyclist guidance behavior, *a priori* definition of a reward function is difficult due to the unknown multivariate relationships among state and action variables that describes the exact trade-offs between different desiderata, and the unknown structure of the reward function itself. This issue is similar to the difficulty in specifying a reward function for robot driving tasks described in (Abbeel and Ng 2004), or in the discrete model developed for inferring a discrete reward function for cyclists' on cycling paths (Mohammed, Sayed, and Bigazzi 2019).

Imitation learning can be used to avoid *a priori* definition of a reward function; imitation learning is the estimation of an agent policy (without a pre-defined reward function) based on observed demonstrations of 'expert' agents. Behavior cloning aims to directly learn a policy from expert demonstrations using a supervised learning approach (Bain and Sammut 1995). A major drawback of behavior cloning is the lack of generality in the learned policy (Ratliff, Silver, and Bagnell 2009). Learning the reward function is another approach, usually called Inverse Reinforcement Learning (IRL) or Inverse Optimal Control (IOC). The major advantage of inductive (IRL) over deductive (behavior cloning) learning approaches is the ability to produce generalizable theories not limited to the training information (Plekhanova 2003).

Reward function learning has several distinct advantages over supervised learning in the context of sequential decision making. The reward in reinforcement learning is defined for all states, allowing an agent to receive a learning signal even from states that are not observed in the training dataset. In contrast, in supervised learning, an agent only receives a score for states included in the labeled observations. In addition, reinforcement learning maximizes the global expected return on a trajectory, rather than the local response to a specific observation. Agents in reinforcement learning have the ability to plan their trajectories considering all future scenarios and may take unusual short-term actions to avoid highly impactful later states.

In summary, a major gap in bicycle microsimulation literature is the limited representation of heterogeneity and uncertainty in cyclist behavior. The current state-of-the-art bicycle microsimulation models use either cellular automata or homogeneous analytic approaches, which do not have the ability to represent the full heterogeneity of cyclist preferences, intentions and interactions. This paper proposes a new methodology for modeling microscopic behavior of cyclists, imitation learning, that allows inference of hidden cyclist preferences which can then be used to predict stochastic cycling behavior in changing environments.

Conceptual and modeling frameworks

The decisions taken by any road user (a cyclist in this case) can be classified into strategic, tactical and operational levels (Michon 1985) – see Figure 1. The strategic level describes high-level decisions about trip planning, such as mode, route, and departure time choices. The operational level describes the microscopic control decisions to operate a bicycle, predominantly pedaling and steering, in addition to finer control aspects such as balance and roll. The middle level (tactical) describes the short-term decisions required to operate the bicycle within the roadway, such as adhering to traffic controls, interacting with other road users, and avoiding obstacles. The three levels of decision-making are connected, meaning that decisions made on the strategic level affect decisions on the tactical level, which in turn affect operational-level decisions. For example, the choice of a specific route that has a right turn at some location will require a tactical decision about when to make the turning maneuver, and consequent steering and pedaling control decisions to execute the maneuver.

Factors that vary between cyclists, such as those related to physical capability and riding equipment, lead to inter-cyclist heterogeneity in decisions. Some factors can also vary for an individual, either over the duration of a single trip or between different trips, creating intra-cyclist decision heterogeneity. Other sources of heterogeneity in decisions are due to environmental variability, such as changing traffic conditions or weather.

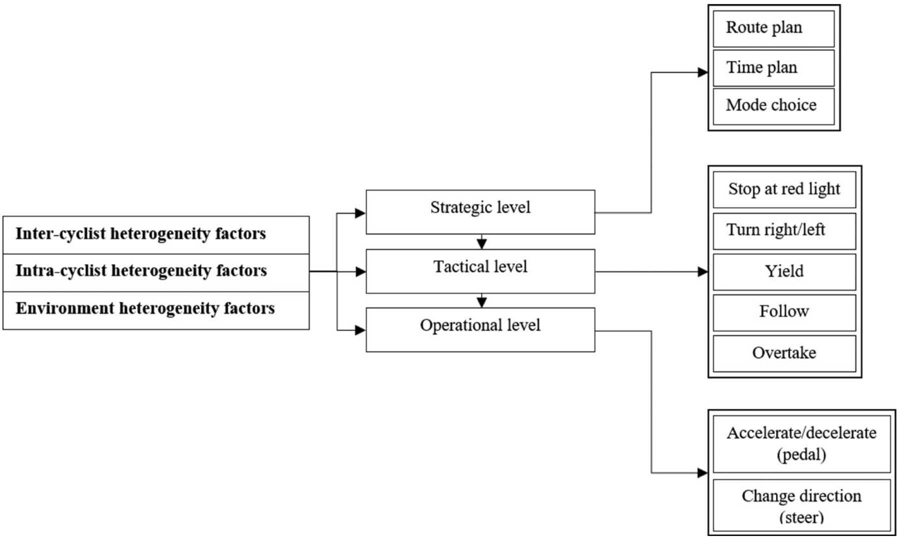

Figure 1. Multiple levels of cyclist on-road decision-making.

Focusing on operational-level decisions for microscopic simulation, we assume cyclists observe attributes of the environment (in particular the movements of other road users) and then make operational control decisions with the intention of executing a tactical-level plan. In building our conceptual framework, we need not only consider sources of heterogeneity that directly affect the operational level decision-making, but we need to consider sources of heterogeneity affecting the higher levels of decision making as well (tactical and strategic). The operational decisions are based on future conditions anticipated by the cyclist, which may not be fully realized due to uncertain environment dynamics. The outcomes of cyclists' operational decisions that are affected by uncertain environment dynamics could change future decisions on the tactical level (deciding to overtake, keep following, keep a certain distance to the center line, etc.) or other tactical decisions that are not explicitly modeled here, such as cyclists waiting an extra block to turn right. Also, the changes in decisions on the tactical or the strategic levels affect the operational level of decisions. Although we are not explicitly modeling the effect of strategic level decisions on the operational level or vice versa, the conceptual framework is needed to scope the modeled behavior and sources of heterogeneity.

In this paper, an agent-based stochastic modeling methodology is used to represent all three levels of heterogeneity. Cyclists are represented by agents, whose states are context-dependent; for example, states of cyclists on bicycle paths have different structures from states of cyclists on roads with mixed traffic. States of cyclists on continuous cycling paths in this paper are defined by state features of speed, position in the path, longitudinal and lateral distances from other cyclists, and speed differences from other cyclists.

The conceptual framework is reflected in the modeling framework formulation in the choice of the modeling approach itself, which allows for capturing the three levels of heterogeneity. In modeling our problem as a stochastic discrete-time continuous state and action Markov decision process. The continuous state and action spaces allow for more realistic modeling of the search space of actions taken by a cyclist facing a certain state.

Also, continuous state and action spaces prevent having gaps in possible action and state variables combinations. Cyclist agents in the model map between their current state and their chosen action using a function called the policy function. This policy function uses an underlying function called the reward function that the agent is trying to maximize. By searching the space for the best action that causes transition to a new state having the highest possible reward given the environment dynamics, the objective of the cyclist agent is being defined and the policy function is formed. This formulation is aiming to mimic the operational decision-making process of cyclists in real life by choosing actions from a continuous search space leading to change in the future state done on fine discrete intervals (1/30th) of a second.

Two types of agents are defined in the model: unconstrained agents, whose operational decisions are not affected by the presence or movements of other road users, and constrained agents, whose decisions are affected by other road users (for example, a cyclist following another cyclist). The decisions of constrained and unconstrained agents are modeled differently. Constrained agents make decisions about their control actions by applying a learned stochastic policy to information about the environment and other agents. The policy seeks to maximize the discounted rewards of uncertain future states for the agent. Unconstrained agents apply a learned stochastic policy responding only to the environment.

Methodology

Data

The data used as demonstration for training and testing of the model are bicycle trajectories extracted from video data recorded at the Burrard Bridge in Vancouver, Canada (Figure 2). Video data with a frame frequency of 30 Hz were obtained over three days, 12–14 April 2016 from 7:00 to 19:00 (29 h total). The weather during the three days was partly cloudy with a stable average temperature of 11°C. The video image included bicycle traffic in a dedicated unidirectional path (without pedestrians) with a grade of +1% in the direction of travel.

Spatiotemporal bicycle trajectories composed of the location, speed, acceleration, and direction of the bicycles at each time step (1/30th of a second) were extracted from the video data by means of computer vision techniques (Saunier and Sayed 2006; Ismail, Sayed, and

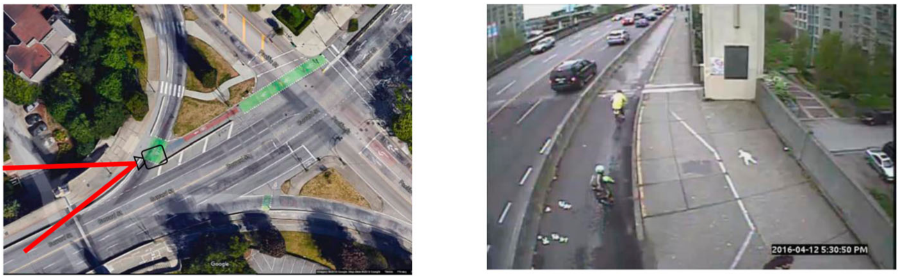

Figure 2. Camera position (left) and field of view (right) for the video data.

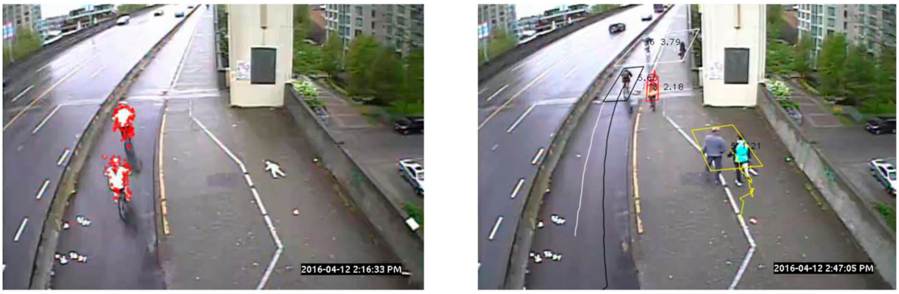

Figure 3. Illustrations of feature detection (left) and object grouping (right).

Saunier 2008). The first step for the computer vision software is camera calibration to convert the two-dimensional video coordinates into three-dimensional real-world coordinates by creating a homography matrix. Due to the significance of the camera calibration step for the accuracy of extracted trajectories, a comprehensive procedure was followed to achieve an acceptable quality of the extracted positional variables. Numerous distances, angles, parallel lines and perpendicular lines were used for the calibration. A validation of the camera calibration was done by comparing speeds of a random sample of cyclists to speeds calculated manually by observing the times it took these cyclists to traverse a section of a certain known length. A mean absolute error of 0.86 m/s between the manually and automatically extracted speeds was observed which was considered sufficiently low to not affect the results. The accuracy of the used computer vision technique was assessed in a previous study (Li et al. 2012), in which a comprehensive accuracy testing was conducted and an R^2 of 0.93 was observed between automatically and manually measured speeds. The software then detects the moving objects by differentiating them from static features using the Kanade-Lucas-Tomasi feature tracker (Lucas and Kanade 1981; Tomasi and Kanade 1994) (Figure 3). The coordinates of the bicycles in each video frame are used to calculate speed, acceleration, and direction. Speed and acceleration profiles are smoothed using the Savitzky–Golay low pass filter (Savitzky and Golay 1964) with a seven time-step (7/30 s) window size. The total number of observed in-interaction trajectories was 2823, in which 2428 were involved in following interactions (following cyclist remained following leading cyclist throughout the observation period) and 395 were involved in overtaking interactions (the originally following cyclist took an overtaking action within the observation period).

Cyclist states are defined by the following variables: x- and y-coordinates in the horizontal plane (m), speed (m/s), and direction angle with respect to the path centerline (degrees). Constrained cyclists are identified in the extracted trajectories as those that co-exist with other cyclists in the same video frames and are in a following position. The video image covered a travel distance of approximately 25 m, so consecutive cyclists with consistently longer spacing were not identified as constrained. The variables used as reward features for constrained cyclists in the imitation learning algorithm are: cyclist speed (v), speed difference from leading cyclist (Δv), longitudinal (D_{long}) and lateral distance (D_{lat}) in path from leading cyclist, path deviation from centerline (D_{cl}), direction angle with respect to path centerline (Φ) and direction angle difference from leading cyclist ($\Delta\Phi$). Illustrations of some of these reward features are provided as $x - y$ plots in Figure 4.

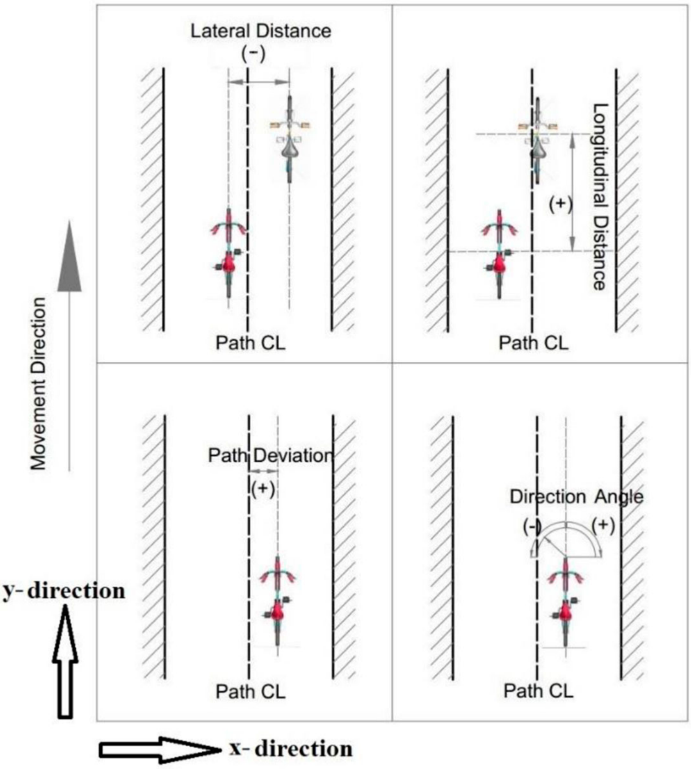

Figure 4. Illustration of some variables used as reward features.

The reward features are the variables that are used as an input to the reward function, which is the basis of the reward function. The mapping from states to actions in the policy function happens by selecting the action that gives the highest reward given some state variables among the continuous search space of possible actions.

Model setup

Control decisions by constrained cyclist agents are modeled as a continuous state-action discrete-time Markov Decision Process (MDP). A model that uses continuous state and action variables is crucial to avoid the exponential increase in parameter space with additional variables or finer discretization. Also, discretization of state and action spaces ignores an infinite number of overlapping state variables and action variables scenarios, which can impede accurate representation of highly detailed cycling behavior. However, the decision was made to use discrete time instead of a continuous time scale because modeling the problem with continuous time would add an unnecessary layer of complexity. Contrary to the discretization of state and action variables, using discrete time scale does not increase the problem dimensionality as the MDP is evaluated at each time step. Also, a high frequency of agent decisions at 1/30th of a second can be assumed as a suitable approximation of a real-world decision-making process.

An MDP consists of a tuple$(\mathcal{S}, \mathcal{A}, \mathcal{T}, \gamma, \mathcal{R})$, where $\mathcal{S}$ is a set of states; $\mathcal{A}$ is a set of actions; $\mathcal{T} = \{P_{sa}\}$ is a set of state transition probabilities; $\gamma \in [0, 1]$ is a discount factor for future rewards; and $\mathcal{R}$is the reward function for a given state $s \in \mathcal{S}$. The cyclist agent MDP has continuous states $(s_1, \ldots, s_t, \ldots, s_T)^T$ and continuous actions $(a_1, \ldots, a_t, \ldots, a_T)^T$ at discrete time t in a sequence of T time steps.

Each cyclist has four state features at each time step: x coordinate, y coordinate, speed v, and direction angle Φ. The total number of state features in the model at each time step depends on the number of agents. For n cyclists existing in a certain time step (one of which will be leading or unconstrained and the others following or constrained), the complete state vector contains $n \times 4$ features: $\{(x_l, y_l, v_l, \Phi_l), (x_{f_1}, y_{f_1}, v_{f_1}, \Phi_{f_1}), (x_{f_2}, y_{f_2}, v_{f_2}, \Phi_{f_2}), \ldots, (x_{f_{n-1}}, y_{f_{n-1}}, v_{f_{n-1}}, \Phi_{f_{n-1}})\}$. Each cyclist has two action features at each time step: acceleration α and yaw rate Θ (rate of change in direction angle). For n cyclists, the complete action vector is thus $\{(\alpha_l, \Theta_l), (\alpha_{f_1}, \Theta_{f_1}), (\alpha_{f_2}, \Theta_{f_2}), \ldots, (\alpha_{f_{n-1}}, \Theta_{f_{n-1}})\}$. The actions of the constrained cyclists are determined by a policy derived from imitation learning, described next; the action of the unconstrained cyclist is determined by a policy derived from a variational autoencoder (described below in a subsequent subsection).

It was decided to use two separate models to predict the behavior of constrained and unconstrained cyclists at each time-step. Constrained cyclist behavior tends to be partially affected by the movement and actions of a leading (unconstrained) cyclist, in addition to other variables related to the environment and individual cyclist characteristics. In contrast, the unconstrained cyclist behavior tends to be more random in nature as cyclists have a wide range of motion variables choices (which are inherently more restricted for constrained cyclists). As a suitable approach for modeling unconstrained cyclists, GAIL was chosen as it learns the hidden stochastic policy function that is a function of relative motion variables and other environment/personal characteristics. Variational Autoencoders was found to be more suitable to modeling unconstrained cyclists as it has the ability to learn a latent distribution of the range of behavioral 'styles' that differ among cyclists.

Constrained agent behavior

An agent's policy is a mapping function between a state $s \in \mathcal{S}$ and the probability of taking a certain action $a \in \mathcal{A}$. The policy function can be written as $\pi_\theta(a|s)$, where θ is a set of parameters for the policy function. In imitation learning, we have a set of M trajectories or expert demonstrations $\mathcal{D} = \{\zeta_1, \ldots, \zeta_m, \ldots, \zeta_M\}$ that are assumed to represent optimal or sub-optimal behavior and are used to estimate the policy and reward function. The goal of policy inference is to find the set of parameters θ that maximizes the likelihood of the observed data (i.e. the cumulative probability the model assigns to the observed trajectories). The expert demonstrations$\mathcal{D}$ are represented by an array of state and action pairs for each observed trajectory: $\zeta = (s_1, a_1, \ldots, s_T, a_T)$. The likelihood of a certain trajectory (ζ) resulting from a policy function π_θ and initial state (s_1) with probability $P(s_1)$ is:

$$L_{\pi_\theta}(\zeta) = L_{\pi_\theta}(s_1, a_1, \ldots, s_T, a_T) = P(s_1)\pi_\theta(a_1|s_1) \prod_{t=2}^{T} \{P(s_t|s_{t-1}, a_{t-1})\pi_\theta(a_t|s_t)\}$$

where $P(s_t|s_{t-1}, a_{t-1})$ is the state transition probability mapping from the state-action pair at one time step (s_{t-1}, a_{t-1}) to the next state s_t (estimated from the observed trajectories).

Reinforcement learning assumes that observed cyclists are following an expert policy, π_E, and that their actions seek to maximize the individual total return, which is the sum of expected discounted rewards along the trajectory. The discounted reward of a future state is represented by $\gamma^t r(s_t)$, where $r(s_t)$ is the single reward associated with visiting state s at time t. The cumulative discounted reward for a trajectory ζ is then $R_\zeta = \sum_{t \in T} \gamma^t r(s_t)$. To infer the expert policy from the observed trajectory data, an optimal policy estimation algorithm would seek to maximize the sum of discounted rewards over the observed trajectories: $\theta^* = \arg\max_\theta E_{\zeta \sim L_{\pi_E}(\zeta)}[R_\zeta]$.

The maximum entropy (ME) principle is used to represent uncertainty in the learning process (Ziebart et al. 2008). In ME the observed trajectories are assumed to represent near-optimal behavior; the agents are assumed to seek an optimal policy, but sub-optimal behavior is possible. A similar representation can be used for the process of humans driving as well (Hamdar, Treiber, and Mahmassani 2008). Relaxing the assumption that the observed trajectory strictly maximizes R_ζ, we instead assume that the probability of a trajectory increases with R_ζ. Using ME, the probability of occurrence for a specific trajectory is a function of the exponential of the aggregate reward: $P(\zeta) = (exp(R_\zeta)/Z)$, where Z is the partition function that represents the aggregate rewards for all possible trajectories. The partition function Z is intractable to calculate, and so it is approximated using importance sampling.

Although $r(s_t)$ is unknown and unobserved, a surrogate reward $\tilde{r}(s_t)$ can be learned from the data, without imposing a structure on the reward function, using a generative adversarial imitation learning (GAIL) procedure. GAIL trains a generator G_θ to perform expert-like behavior by rewarding it for 'deceiving' a classifier or discriminator D_ψ that is trained to discriminate between the generated and observed state-action pairs. Consider a set of simulated trajectory data ζ_θ sampled from G_θ and a set of expert trajectories ζ_E sampled from the expert dataset, $\mathcal{D}$.

For a neural network D_ψ parameterized by ψ, the GAIL discriminator loss function is given by $\mathcal{L}(D_\psi) = \mathbb{E}_{\zeta_E \sim \mathcal{D}}[-\log D_\psi(\zeta_E)] + \mathbb{E}_{\zeta_\theta \sim G_\theta}[\log(1 - D_\psi(\zeta_\theta))]$. The first term of the discriminator's loss function is the effect of correctly classifying an expert trajectory, $D_\psi(\zeta_E)$; the second term is the effect of wrongfully classifying a generated trajectory, $D_\psi(\zeta_\theta)$. The minimum value of the loss function occurs when the discriminator becomes indifferent to trajectories sampled from expert data versus from the generator. The GAIL generator loss function is similar to that of the discriminator, but is minimized when the discriminator is confused in classifying the generated trajectories as expert ones: $\mathcal{L}(G_\theta) = \mathbb{E}_{\zeta_\theta \sim G_\theta}[-\log D_\psi(\zeta_\theta)]$.

The discriminator's classifier is trained using binary logistic regression, where $P_{\pi_E}(\zeta)$ denotes the probability of occurrence of trajectory ζ under the expert policy π_E, and $P_{\pi_\theta}(\zeta)$ denotes the probability of occurrence under the generator policy π_θ: $D_\psi(\zeta) = (P_{\pi_E}(\zeta)/P_{\pi_E}(\zeta) + P_{\pi_\theta}(\zeta))$. Substituting the ME definition of the probability of occurrence for expert trajectories, the discriminator becomes: $D_\psi(\zeta) = (exp[R(\zeta)]/exp[R(\zeta)] + Z \cdot P_{\pi_\theta}(\zeta))$. The generator's policy is modeled as a neural network with rectified linear units (ReLU) that captures non-linearity in the policy function. The policy is trained by backpropagation using minibatch gradient descent; positive examples are sampled from ζ_E

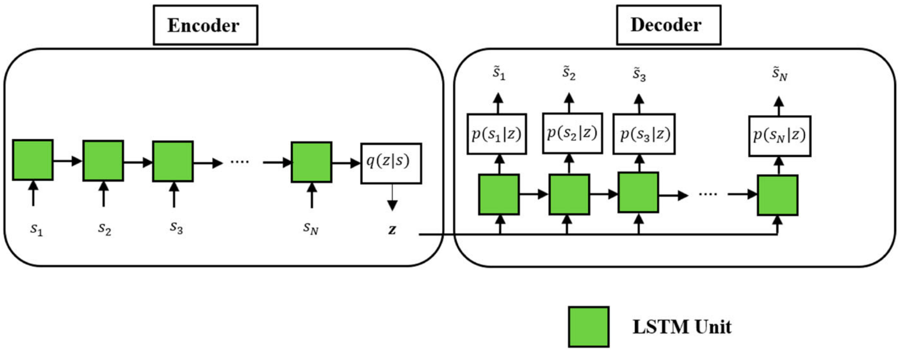

Figure 5. Illustration of the LSTM-variational autoencoder architecture.

and negative examples are sampled from rollouts generated by interactions of π_θ with the simulation environment.

To estimate π_θ, a surrogate reward function is formulated as: $\tilde{r}(s_t; \psi) = -\log(1 - D_\psi (s_t, a_t))$. After performing the rollout with a given set of policy parameters θ, surrogate rewards $\tilde{r}(s_t; \psi)$ are calculated and trust region policy optimization (Schulman 2015) is used to perform the policy update. Although $\tilde{r}(s_t; \psi)$ may be different from the true reward function optimized by experts, it can be used to drive π_θ into regions of the state-action space similar to those explored by π_E.

Unconstrained agent behavior

A variational autoencoder (VAE) is used to model the unconstrained agent behavior. VAE trains two neural networks, the 'encoder' and the 'decoder'. The encoder takes as input the observed trajectories (sequence of states, s_t) of unconstrained cyclists and its output is a hidden latent representation space $\boldsymbol{z}$. The latent dimension space $\boldsymbol{z}$ is a stochastic Gaussian distribution from which samples can be drawn. The encoder process is denoted as $q_\vartheta(z|s)$, where ϑ is the encoder network parameters. The decoder process acts in reverse to the encoder, in which the input is the latent variable representation $\boldsymbol{z}$ and the output is a reconstruction of unconstrained cyclist trajectories. The decoder network is denoted as $p_\phi(s|z)$, where ϕ is the decoder network parameters. The structure of the VAE is illustrated in Figure 5.

The VAE encoder and decoder are both defined by long-short-term-memory (LSTM) units that are the same in number as the length of the input trajectory. This LSTM architecture allows for capturing the interdependence between sequential observations (short term) and the overall interdependence between all the points constituting the trajectory (long term). The latent variable layer contains the Gaussian distribution parameters (mean μ and standard deviation σ). The neural network is mirrored to re-estimate the observed trajectory states at each time step ($\tilde{s}_t$), with the latent variable z calculated as $z = \mu + \sigma * \epsilon, \epsilon \sim \mathcal{N}(0, 1)$. This representation allows generation of samples from the network (after training) simply by sampling a cluster and a value for the random normal variable ϵ.

Model testing

Cross-validation testing is undertaken using a testing dataset of 20% of the observed trajectories, randomly selected and removed from training. Simulations of the testing trajectories are made using the estimated model, with starting states equivalent to the test data. Aggregate distributions of both model variables (longitudinal distance, lateral distance, speed, speed difference, direction angle, direction angle difference, and deviation from the centerline) and emergent variables (jerk and acceleration) are compared between simulated and observed trajectories to assess the accuracy and emergence of the model, and its ability to represent heterogeneous behavior.

A two-dimensional Kolmogorov–Smirnov test is used to compare multidimensional parameter distributions (Fasano and Franceschini 1987), with a 95% confidence threshold to reject the null hypothesis that the density distributions of paired state features in the simulated trajectories are drawn from the same distributions as the observed data. Kullback-Leibler divergence (KL divergence) is also used to compare bivariate distributions of state and action features between simulated and observed trajectory data. The KL divergence between observed $p(x)$ and simulated $q(x)$ data is a measure of the information lost when $q(x)$ is used to approximate $p(x)$, where x is the variable of interest, calculated:

$$D_{KL}(p(x)||q(x)) = \int_{-\infty}^{\infty} p(x) \log \frac{p(x)}{q(x)}$$

The information loss is measured in bits, a measurement unit for a distribution's entropy.

The model is also evaluated by comparison to two other cyclist simulation models from the literature, selected for their relevance to the modeling scope and representation of alternative approaches to cyclist simulation (Jiang et al. 2016; Zhao et al. 2013). These models are implemented and calibrated to the same dataset, and then the simulated trajectories of all three models and observed cyclists are compared. The details of the comparison models, calibration, and performance are reported in the penultimate section of the paper, following the presentation of the proposed model results in the next section.

Results

The plots in Figure 6 give one realization of the estimated reward values over reward features (left), along with histograms of observed frequency of feature values in the dataset (right). Higher reward values imply preferred states for cyclists, all else equal. The relationship between reward values and longitudinal distance shows a preference for longitudinal (following) distance peaking around 5 m. There is also a preference for lateral distances near zero (following directly behind the leading cyclist), and to be following toward the right versus toward the left (positive rather than negative lateral distance, at a given magnitude). The reward over varying deviation from the path centerline also clearly indicates a preference for staying on the right side of the path (as expected). A direction angle near zero (following the path) is preferred, with a preference to turn left (toward the centerline) rather than turn right (toward the right edge), based on the reward values at positive and negative direction angles of similar magnitude.

Inferred speed preference peaks above 3 m/s ($\sim$ 11 kph), despite lower cycling speeds being more common due the hindering presence of other cyclists. Reward decreases much

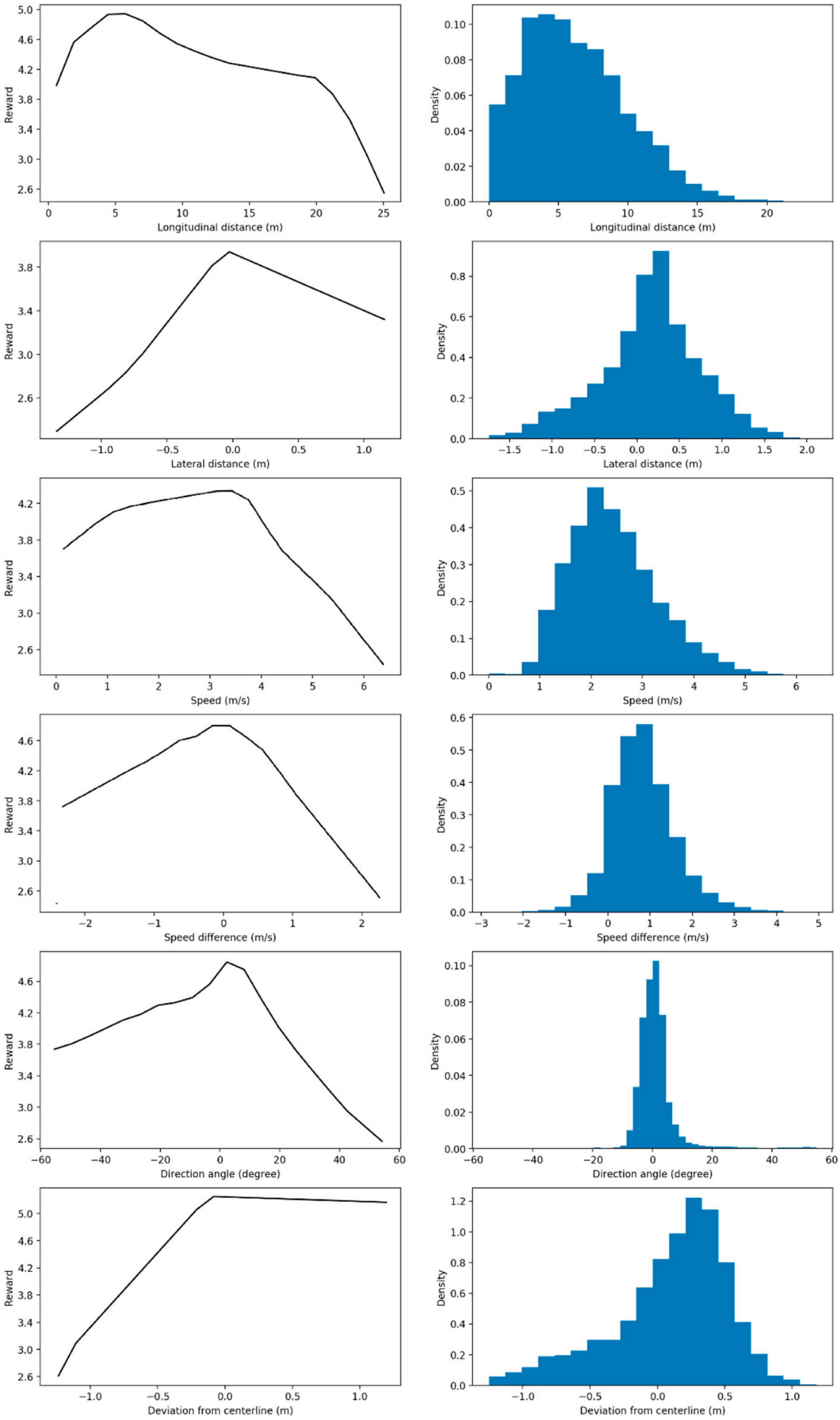

Figure 6. Reward function visualization with univariate reward feature distributions.

Table 1. Cross-validation results comparing observed and simulated distributions.

Variable	KL divergence	Loss in information bits
Longitudinal distance	0.80	8%
Lateral distance	1.51	14%
Speed	0.96	9%
Speed difference	0.45	6%
Direction angle	1.67	15%
Direction angle difference	1.22	12%
Centerline deviation	0.38	5%
Acceleration range (per meter)	0.35	6%

faster toward higher speeds than toward lower speeds, likely because the former requires more pedaling power, whereas the latter requires less and so is easier to accomplish. Cyclists also prefer to match the speed of a leading cyclist (speed difference near zero), but would prefer to be going slower (negative difference) rather than faster, which requires them to decelerate or overtake. A preference for low speed differences to leading cyclists is similar to results reported in Ma and Luo (2016). Other inferred preferences are not comparable to previous studies which only predicted longitudinal motion dynamics (Twaddle and Grigoropoulos 2016; Zhao and Zhang 2017).

Cross-validation distribution analysis results are given in Table 1. KL divergence is low and the loss in information for the simulated versus observed data range from 5% to 14% across variables. Figure 7 gives observed and simulated bivariate density plots of state variables. Visual inspection reveals resemblance between the observed and simulated state variable distributions. For example, the speed difference tends to decrease to zero as longitudinal distance approaches zero. Two-dimensional Kolmogorov–Smirnov tests fail to distinguish between observed and simulated distributions at $p < 0.05$ for all variable pairs.

To illustrate the modeled heterogeneity in cycling paths, Figure 8 shows an arbitrary observed trajectory and the distribution of 1000 trajectories simulated from the same initial state and environment dynamics. The observed trajectory lies within the simulated trajectory distribution. Variation in the simulated paths arises from the stochasticity in the agent policy and state transitions. Stochasticity in the model allows inspection of how heterogeneity varies along the path.

Overtaking maneuvers are the most dynamic behavior to model on unidirectional cycling paths. Figure 9 gives a visual comparison of observed and simulated overtaking patterns as heatmaps of the relative locations of overtaking cyclists with respect to the leading cyclist's position. Both observed and simulated heatmaps show a higher density of overtaking maneuvers executed from the left of the originally leading cyclist. In addition, an 'influence area' around the originally leading cyclist is observed as a lack of observations near the origin in both datasets, at about ± 1.6 m longitudinally and ± 0.4 m laterally (increasing slightly at the overtaking position of $dy = 0$). These result shows that without explicitly including cyclist dimensions in the model, the agents faithfully reproduce the operating space of observed cyclists.

To get a sense of how the modeled behavior during overtaking maneuvers compares to literature, the results are compared to Khan and Raksuntorn (2001), in which motion variables from video data of cyclists in following and overtaking interactions are extracted. In

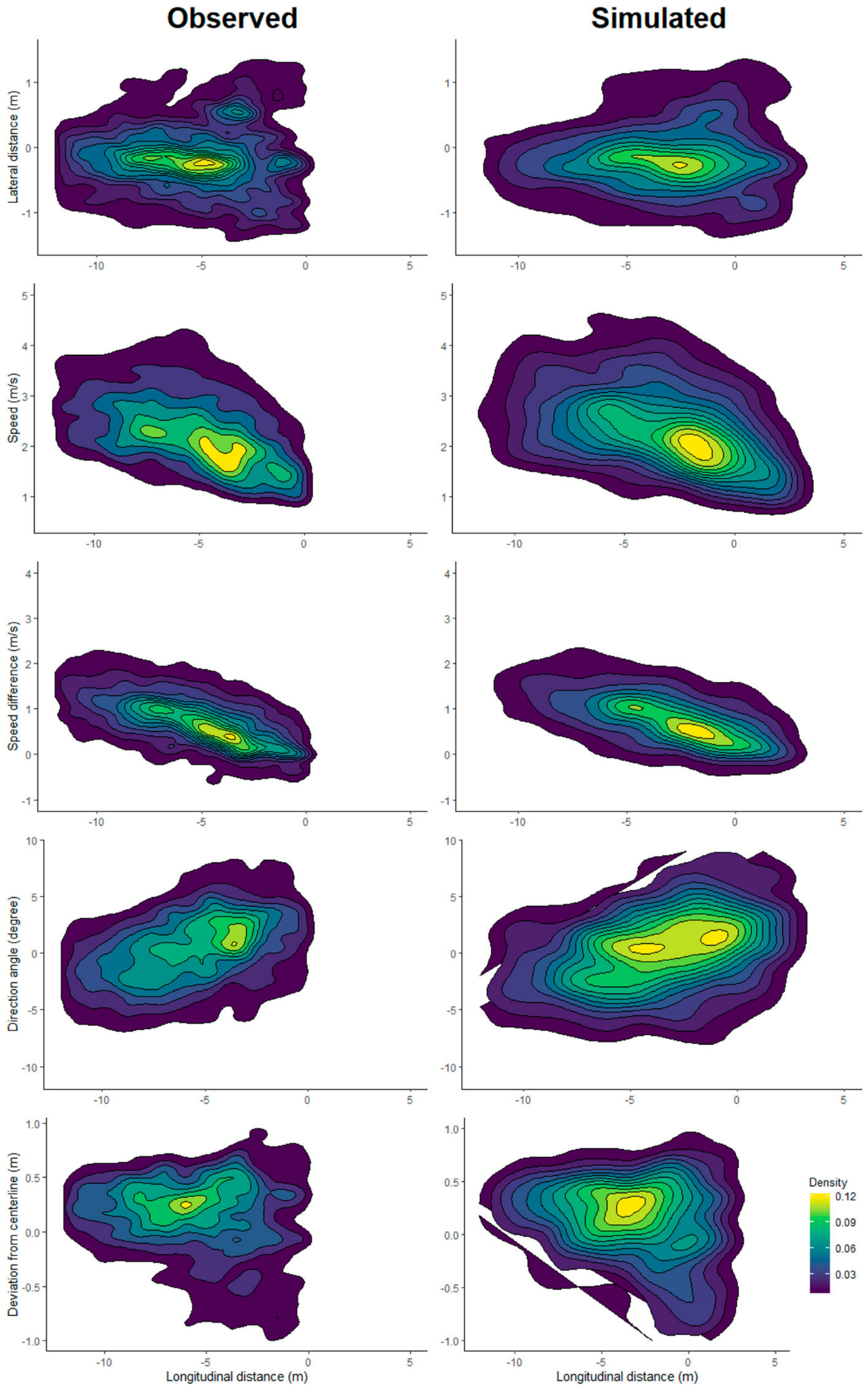

Figure 7. Bi-variate density plots of observed and simulated trajectories.

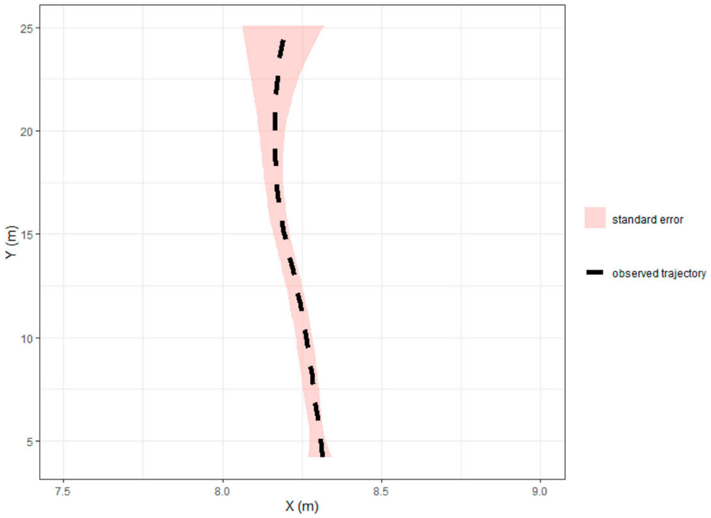

Figure 8. Example of the intra-cyclist variability captured by the model.

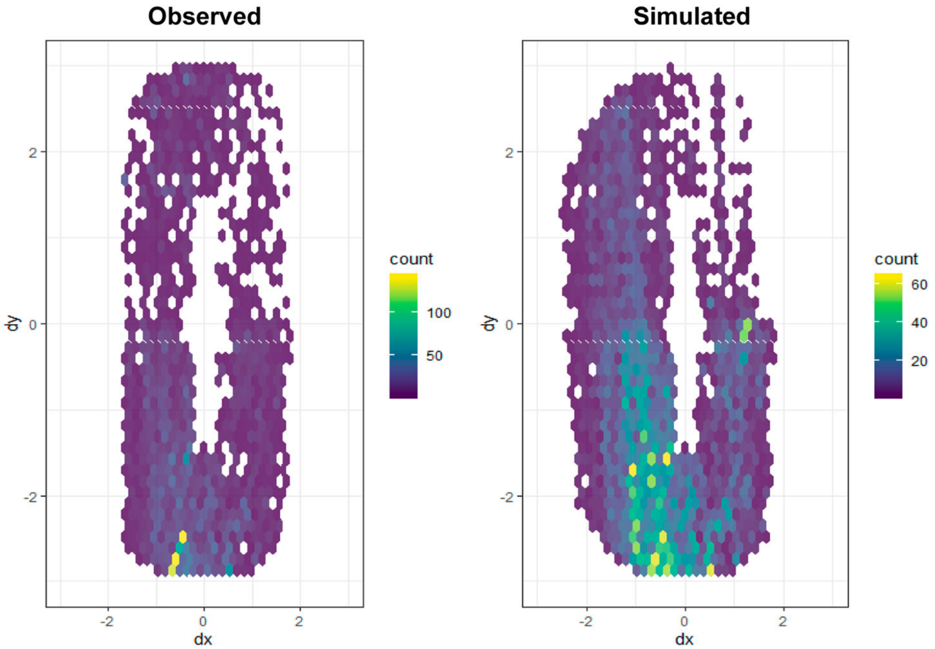

Figure 9. Locations of overtaking cyclists in observed and simulated data (in m, with respect to the cyclist being overtaken at $(\boldsymbol{dx}, \boldsymbol{dy}) = (0, 0)$.

that study, the average lateral distance during overtaking was 1.78 m compared to about 1 m in our study (Figure 9). The difference in lateral distance could be due to different cycling path widths (3 m in the Khan and Raksuntorn 2001 study versus 2.2 m in our study).

Variable density functions in Figure 7 are compared against a similar study (Gavriilidou et al. 2019) that used an analytical approach to assess how the preferences of cyclists toward taking certain actions change with relative motion variables. In that study preference for lower deviation from the centerline was observed and implemented in their model, which resembles our model predictions (Figure 7). Also, in another recent study (Paulsen, Rasmussen, and Ank 2019), in which the authors were trying to establish the effect of bicycle traffic congestion on microscopic cycling behavior, it was observed that cyclists prefer to move at higher speeds and speed differences when their longitudinal distances are higher, which can also be seen in Figure 7.

Comparison to other models

To further assess the performance and benefit of the imitation learning model presented in this paper, comparisons are made to two other cyclist simulation models from the literature. These two models were selected because they have similar scope to the proposed model and represent alternative state-of-the-art microsimulation methods. The first model (Jiang et al. 2016) is based on the well-known cellular automata (CA) approach developed by Nagel and Schreckenberg (2002). The model was originally developed from an experiment in which multiple cyclists navigated an oval course. The second model (Zhao et al. 2013) also uses the CA approach to model bicycle following and overtaking interactions, developed from video data of a separated bicycle path. Unlike the first model which only represents unidimensional following behavior, the second model simulates both longitudinal and lateral movements.

The parameters of both comparison models were calibrated to the study dataset. The Jiang et al. (2016) model predicts the longitudinal position of a cyclist at each 1-second time step using the instantaneous speed. The instantaneous speed is updated each time step, influenced by the longitudinal distances between successive cyclists, with a randomization parameter. The calibration parameters are the maximum cyclist speed $v_{\max}$, the operating distance between cyclists d_{op}, and a third parameter d_c reflecting the effect of a leading cyclist's distance on the following cyclist's decisions. The Zhao et al. (2013) model updates both longitudinal and lateral positions of cyclists at each time step from instantaneous longitudinal and lateral speeds, which are influenced by the relative longitudinal and lateral positions of the following and leading cyclists. The model also has a randomization component to the speed updating equations. There is just one parameter to calibrate, which is the maximum speed $v_{\max}$.

For this comparison, the parameters in the Jiang et al. (2016) model were calibrated to the study path data set by enumerating values in appropriate ranges of the calibration parameters and then calculating the root mean squared error (RMSE) between the x- and y- locations (m) in the simulated vs. observed data for each combination of parameter values. The calibration parameter values that yielded the lowest RMSE were $v_{\max} = 5.2$ m/s, $d_{op} = 6$ m, and $d_c = 2$ m. The selected value of the randomization parameter p was 0.8, based on the original paper. The parameter of the Zhao et al. (2013) model was calibrated similarly with the result of a calibration value of $v_{\max} = 4.8$ m/s. The selected value of the randomization parameter p for the Zhao et al. model was 0.45, based on the original paper.

Figure 10 gives the distributions of motion variables for the observed data and for the simulated output of the imitation learning model proposed in this paper and the two calibrated comparison models (at 1-second intervals, $N = 5645$). The distributions of longitudinal distance, speed, speed difference and lateral distance from each model are based

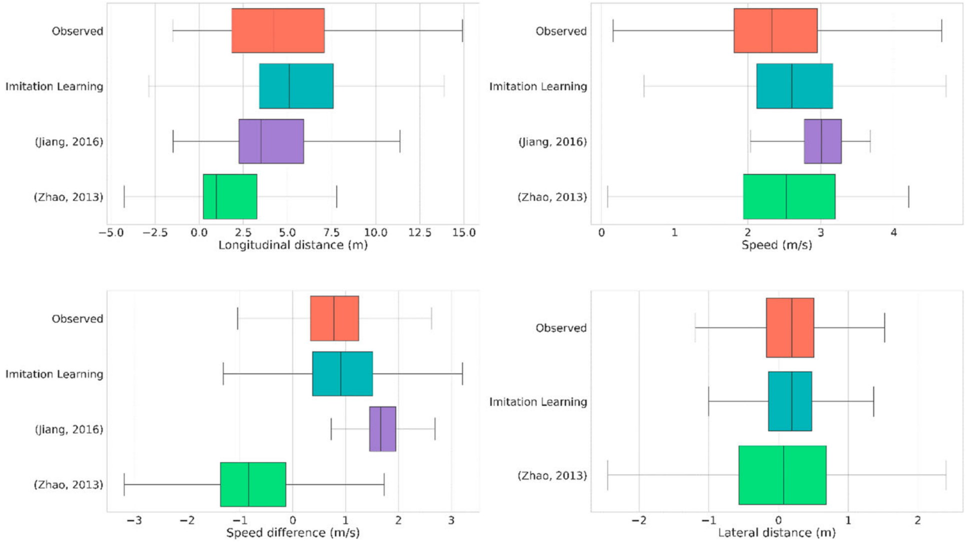

Figure 10. Motion variable distributions from observation and the three simulation models.

on a single realization (simulation run) for all interactions in the study data set. Overall, the imitation learning model simulates the observed distributions of motion variables more accurately than the comparison models. The Zhao et al. model has the poorest performance for longitudinal distance, and the Jiang et al. model has the poorest performance for speed. Both comparison models poorly represent the observed speed differences between leading and following cyclists. The Zhao et al. model produces too wide a range of lateral distances, while the Jiang et al. model does not represent the lateral dimension at all.

In addition to these differences in model performance, there are conceptual advantages of the proposed imitation learning model. First, the proposed model represents the behavior of cyclists in unconstrained and constrained states, and allows flexible transitions between those states. In contrast, the comparison models do not explicitly model unconstrained behavior and instead assume constant or randomly fluctuating speed for 'free-flow' cyclists, which further affects the simulated behavior of the following or overtaking cyclists. The imitation learning model also explicitly represents heterogeneity in both the environment and cyclist behavior. In contrast, the comparison models introduce stochasticity simply with a random parameter in the speed updating process, which cannot represent the systematic differences in physical, psychological and other characteristics of cyclists.

To compare the stochastic performance of each model, 120 simulation runs for the entire study data set were executed using all three models (imitation learning, Jiang et al., and Zhao et al.). Figure 11 gives the resulting performance in terms of the mean absolute value of the mean error (MAE) of the longitudinal distance variable for each simulated trajectory, compared to the observed data. Each point in the figure gives the MAE for a single model run on the entire dataset (i.e. the distributions in Figure 10 would yield a single point in each column); the jitter plot randomly positions points in the horizontal direction to prevent overlapping and improve visual interpretability. The imitation learning model has the most consistently high precision, with an MAE under 3 m for all 120 runs, compared to almost

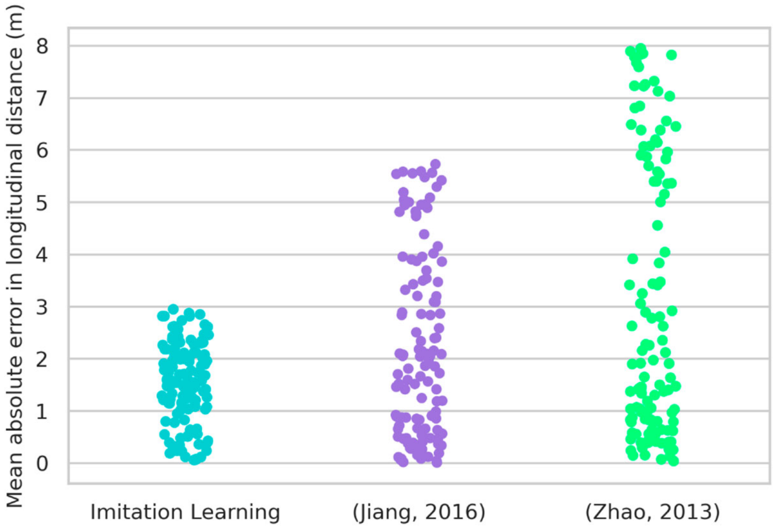

Figure 11. Distributions of MAE over 120 runs of the entire data set by each model.

6 m for the Jiang et al. model and almost 8 m for the Zhao et al. model. The difference in stochastic performance may be due to an overly simple representation of heterogeneity in the comparison models.

Conclusion

This paper proposes and tests the applicability of generative adversarial imitation learning for estimating continuous, non-linear, and stochastic policies and reward functions for cyclists in an agent-based simulation model. In addition these advancements, the model is novel in predicting both longitudinal and lateral motion dynamics of interacting cyclists. Cross-validation of the simulation model indicates realistic generation of modeled and emergent variable distributions. The proposed model also outperforms two cyclist simulation models from the literature representative of state-of-the-art alternative approaches.

A core contribution of this paper is the development of the novel approach to agent policy estimation, with greater flexibility and fewer *a priori* assumptions than existing approaches. This research aims to advance microscopic modeling of cyclists by following an intelligent agent approach with a non-linear stochastic decision-making mechanism. The model differentiates between two distinct types of cyclists (constrained and unconstrained). Constrained cyclist behavior and decision making are dependent on cyclist personal characteristics, environment variables, motion variables of the constrained (following) cyclist, and relative motion variables between the following and the leading cyclist. Unconstrained cyclist behavior is only dependent on the cyclist's personal characteristics, environment variables, and own motion variables. The paper follows a generative modeling approach (as opposed to discriminative approaches) in modeling interacting cyclists' trajectories. Generative models are concerned with learning the internal multivariate distributions of model variables, while discriminative approaches identify the boundaries between behavior schemes.

The key limitation in the presented work is model testing in a single case study location. The model application used short-distance observed trajectory data (around 25 m), which precluded analysis of longer spatial patterns. It is a challenge to collect detailed trajectory data over long distances from existing computer vision techniques. This limitation could be addressed in future research by creating a longer-coverage dataset from two or more synchronized cameras. Despite this limitation in the observed length of trajectories, the imitation learning approach that uses memoryless MDP representation allows for making inferences about behavior even from short sequences of observations. The memoryless property means that agents use the estimated policies to take actions only based on the current state. Also, the large number of trajectories allows for observing wide distributions of motion variables that are used in the model (Figure 6).

Another limitation related to the video data is that it may not contain much variability in some of the factors expected to influence cyclist behavior. Examples include cyclist sociodemographics, variables related to cycling effort such as road grade, wind, and mass, as well as factors relating to trip purpose and length, weather, and time of day. In this study, these factors would likely manifest in the variability of the stochastic policy estimation.

Related to these limitations, essential future research tasks include applying and testing the model in other locations, ideally with longer trajectories and augmented cyclist attributes (both of which would require enhanced data collection methods). Applying the model in other locations will allow further evaluation of different levels of model uncertainty and generalizability, with investigation of variability at the cyclist and location levels. Due to the unique nature of agent-based models, pattern-based validation methods should also be considered for further model validation. Finally, there are several key paths for future model development: most importantly extension to two-way paths and additional agent types representing pedestrians and other 'multi-use path' users such as people on skateboards, e-scooters, and other devices. By enhancing the representation of cyclist heterogeneity in agent-based simulations, we believe that the proposed approach is a significant methodological advancement toward more nuanced and sensitive analysis tools in active transportation.

Acknowledgement

The authors would like to thank the BITSAFS lab team for providing support in data extraction and processing, and the REACT lab team regarding methodology conceptualization and formulation and results visualization. H. Mohammed: Data curation; Investigation; Methodology; Writing - original draft; Writing - review & editing; T. Sayed: Conceptualization; Methodology; Software; Resources; Supervision; Writing - review & editing; A. Bigazzi: Conceptualization; Methodology; Resources; Supervision; Writing - review & editing

Disclosure statement

No potential conflict of interest was reported by the authors.

ORCID

Hossameldin Mohammed http://orcid.org/0000-0001-5093-0204
Tarek Sayed http://orcid.org/0000-0001-8797-0541
Alexander Bigazzi http://orcid.org/0000-0003-2253-2991

References

Abbeel, P., and A. Ng. 2004. "*Apprenticeship Learning via Inverse Reinforcement Learning*." *Proceedings of the twenty-first international conference on machine learning*.

Abdou, M., L. Hamill, and N. Gilbert. 2012. *Designing and Building an Agent-Based Model. Agent-Based Models of Geographical Systems*. Dordrecht: Springer.

Aghabayk, K., M. Sarvi, and Y. William. 2015. "A State-of-the-art Review of car-Following Models with Particular Considerations of Heavy Vehicles." *Transport Reviews* 1 (35): 82–105.

Bain, M., and C. Sammut. 1995. A Framework for Behavioral Cloning. *Machine intelligence agents*.

Baster, B., J. Duda, A. Maciol, and B. Rebiasz. 2013. "*Rule-based Approach to Human-Like Decision Simulating in Agent-Based Modeling and Simulation*." System Theory, control and Computing (ICSTCC) 17th International Conference.

de Oliveira, Í. 2017. "Analyzing the Performance of Distributed Conflict Resolution among Autonomous Vehicles." *Transportation Research Part B: Methodological* 96: 92–112.

Djavadian, S., and C. Joseph. 2017. "An Agent-Based day-to-day Adjustment Process for Modeling 'Mobility as a Service'with a two-Sided Flexible Transport Market." *Transportation Research Part B: Methodological* 104: 36–57.

Fasano, G., and A. Franceschini. 1987. "A Multidimensional Version of the Kolmogorov–Smirnov Test." *Monthly Notices of the Royal Astronomical Society* 225 (1): 155–170.

Gavriilidou, A., W. Daamen, Y. Yua, and S. Hoogendoorn. 2019. "Modelling Cyclist Queue Formation Using a two-Layer Framework for Operational Cycling Behaviour." *Transportation Research Part C: Emerging Technologies* 105: 468–484. https://doi.org/10.1016/j.trc.2019.06.012.

Gould, G., and A. Karner. 2009. "Modeling Bicycle Facility Operation: Cellular Automaton Approach." *Modeling Bicycle Facility Operation: Cellular Automaton Approach* 2140: 157–164.

Hamdar, S., M. Treiber, and H. Mahmassani. 2008. "Modeling Driver Behavior as Sequential Risk-Taking Task." *Transportation Research Record* 2088: 208–217. 10.3141/2088-22.

Heinen, E., V. Bert, and M. Kees. 2010. "Commuting by Bicycle: an Overview of the Literature." *Transport Reviews* 30 (1): 59–96.

Hussein, M., and T. Sayed. 2017. "A bi-Directional Agent-Based Pedestrian Microscopic Model." *Transportmetrica A: Transport Science* 13 (4): 326–355.

Ismail, K., T. Sayed, and N. Saunier. 2008. "A Methodology for Precise Camera Calibration for Data Collection Applications in Urban Traffic Scenes." *Canadian Journal of Civil Engineering* 40 (1): 96–104.

Jennings, N. 2000. "On Agent-Based Software Engineering." *Artificial Intelligence* 117 (2): 277–296.

Jia, B., X.-G. Li, R. Jiang, and Z.-Y. Gao. 2007. "Multi-value Cellular Automata Model for Mixed Bicycle Flow." *The European Physical Journal B* 56 (3): 247–252.

Jiang, R., M. Hu, Q. Wu, and W. Song. 2016. "Traffic Dynamics of Bicycle Flow: Experiment and Modeling." *Transportation Science* 51 (3): 998–1008. https://doi.org/10.1287/trsc.2016.0690.

Jiang, R., B. Jia, and Q.-S. Wu. 2004. "Stochastic Multi-Value Cellular Automata Models for Bicycle Flow." *Journal of Physics A: Mathematical and General* 37: 6.

Khan, S., and W. Raksuntorn. 2001. "Characteristics of Passing and Meeting Maneuvers on Exclusive Bicycle Paths." *Transportation Research Record: Journal of the Transportation Research Board* 1776: 220–228.

Li, S., T. Sayed, M. Zaki, G. Mori, F. Stefanus, B. Khanloo, and N. Saunier. 2012. "Automated Collection of Pedestrian Data Through Computer Vision Techniques." *Transportation Research Record* 2299: 121–127. https://doi.org/10.3141/2299-13.

Liang, X., M. A. Baohua, and X. U. Qi. 2012. "Psychological-physical Force Model for Bicycle Dynamics." *Journal of Transportation Systems Engineering and Information Technology* 12 (2): 91–97.

Liu, J., and X. Zhou. 2016. "Capacitated Transit Service Network Design with Boundedly Rational Agents." *Transportation Research Part B: Methodological* 93: 225–250.

Lucas, B. D., and T. Kanade. 1981. An Iterative Image Registration Technique with an Application to Stereo Vision. *International Joint Conference on Artificial Intelligence*, 674–679.

Ma, X., and D. Luo. 2016. "Modeling Cyclist Acceleration Process for Bicycle Traffic Simulation Using Naturalistic Data." *Transportation Research Part F: Traffic Psychology and Behaviour* 40: 130–144.

Michon, J. A. 1985. A Critical View of Driver Behavior Models: What Do We Know, What Should We Do? *In Human behavior and traffic safety*, 485–524. Boston, MA: Springer.

Mohammed, H., A. Y. Bigazzi, and T. Sayed. 2019. “Characterization of Bicycle Following and Overtaking Maneuvers on Cycling Paths.” *Transportation Research Part C: Emerging Technologies* 98: 139–151.

Mohammed, H., T. Sayed, and A. Bigazzi. 2019. “*Toward Agent-Based Microsimulation of Cyclist Following Behavior: Estimation of Reward Function Parameters Using Inverse Reinforcement Learning*.” *98th Annual Meeting of the transportation research Board*, Washington D.C.

Nagel, K., and M. Schreckenberg. 2002. “Cellular Automata Approach to Three-Phase Traffic Theory.” *Physica A: Mathematical and General* 35 (47): 9971–10013.

Paulsen, M., T. Rasmussen, and O. Ank. 2019. “Fast or Forced to Follow: A Speed Heterogeneous Approach to Congested Multi-Lane Bicycle Traffic Simulation.” *Transportation Research Part B: Methodological* 127: 72–98. https://doi.org/10.1016/j.trb.2019.07.002.

Plekhanova, V. 2003. *Intelligent Agent Software Engineering*. Hershey, PA: IGI Global.

Ratliff, N., D. Silver, and A. Bagnell. 2009. “Learning to Search: Functional Gradient Techniques for Imitation Learning.” *Autonomous Robots* 27 (1): 25–53.

Saunier, N., and T. Sayed. 2006. “*A Feature-Based Tracking Algorithm for Vehicles in Intersections*.” *Third Canadian Conference on computer and robot vision*, IEEE.

Savitzky, A., and M. J. Golay. 1964. “Smoothing and Differentiation of Data by Simplified Least Squares Procedure.” *Analytical Chemistry* 36 (8): 1627–1639.

Schulman, J. L. 2015. “*Trust Region Policy Optimization*.” *International conference on machine learning*.

Tang, T.-Q., Y.-X. Rui, J. Zhang, and H.-Y. Shang. 2018. “A Cellular Automation Model Accounting for Bicycle’s Group Behavior.” *Physica A: Statistical Mechanics and its Applications* 492: 1782–1797.

Taylor, D. B., and H. S. Mahmassani. 1998. *Behavioral Models and Characteristics of Bicycle-Automobile Mixed-Traffic: Planning and Engineering Implications*. Austin: University of Texas.

Tomasi, C., and T. Kanade. 1994. Detection and Tracking of Point Features. *IEEE Conference on computer vision and pattern Recognition*, 593–600.

Twaddle, H., and G. Grigoropoulos. 2016. “Modeling the Speed, Acceleration, and Deceleration of Bicyclists for Microscopic Traffic Simulation.” *Transportation Research Record: Journal of the Transportation Research Board* 2587: 8–16.

Twaddle, H., T. Schendzielorz, and O. Fakler. 2014. “Bicycles in Urban Areas: Review of Existing Methods for Modeling Behavior.” *Transportation Research Record: Journal of the Transportation Research Board* 2434: 140–146.

Wolfram, S. 1983. “Statistical Mechanics of Cellular Automata.” *Reviews of Modern Physics* 55: 3.

Wolfram, S. 1985. “Twenty Problems in the Theory of Cellular Automata.” *Physica Scripta* T9: 170–183. https://doi.org/10.1088/0031-8949/1985/t9/029.

Xue, S., B. Jia, R. Jiang, X. Li, and J. Shan. 2017. “An Improved Burgers Cellular Automaton Model for Bicycle Flow.” *Physica A: Statistical Mechanics and its Applications* 487: 164–177.

Zhao, D., W. Wang, C. Li, Z. Li, P. Fu, and X. Hu. 2013. “Modeling of Passing Events in Mixed Bicycle Traffic with Cellular Automata.” *Transportation Research Record* 2387: 26–34. https://doi.org/10.3141/2387-04.

Zhao, Y., and H. Zhang. 2017. “A Unified Follow-the-Leader Model for Vehicle, Bicycle and Pedestrian Traffic.” *Transportation Research Part B: Methodological* 105: 315–327.

Ziebart, B., A. L. Maas, A. Bagnell, and A. K. Dey. 2008. “*Maximum Entropy Inverse Reinforcement Learning*.” *Proceedings of the Twenty-third AAAI Conference on Artificial Intelligence*.