ANNUAL REVIEWS

Annual Review of Control, Robotics, and Autonomous Systems

Planning and Decision-Making for Autonomous Vehicles

Wilko Schwarting,[1] Javier Alonso-Mora,[2] and Daniela Rus[1]

[1]Computer Science and Artificial Intelligence Laboratory, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, USA; email: wilkos@csail.mit.edu, rus@csail.mit.edu

[2]Department of Cognitive Robotics, Delft University of Technology, 2628 Delft, The Netherlands; email: j.alonsomora@tudelft.nl

Annu. Rev. Control Robot. Auton. Syst. 2018. 1:187–210

First published as a Review in Advance on January 12, 2018

The *Annual Review of Control, Robotics, and Autonomous Systems* is online at control.annualreviews.org

https://doi.org/10.1146/annurev-control-060117-105157

Keywords

autonomous vehicles, intelligent vehicles, decision-making, motion planning, artificial intelligence, verification, fleet management

Abstract

In this review, we provide an overview of emerging trends and challenges in the field of intelligent and autonomous, or self-driving, vehicles. Recent advances in the field of perception, planning, and decision-making for autonomous vehicles have led to great improvements in functional capabilities, with several prototypes already driving on our roads and streets. Yet challenges remain regarding guaranteed performance and safety under all driving circumstances. For instance, planning methods that provide safe and system-compliant performance in complex, cluttered environments while modeling the uncertain interaction with other traffic participants are required. Furthermore, new paradigms, such as interactive planning and end-to-end learning, open up questions regarding safety and reliability that need to be addressed. In this survey, we emphasize recent approaches for integrated perception and planning and for behavior-aware planning, many of which rely on machine learning. This raises the question of verification and safety, which we also touch upon. Finally, we discuss the state of the art and remaining challenges for managing fleets of autonomous vehicles.

1. INTRODUCTION

Autonomous vehicles will reduce the number of road fatalities, give our parents and grandparents greater independence in their retirement, and give us the ability to go anywhere, anytime. In a single year, Americans drive nearly 3 trillion miles (1), which translates into many hours spent in traffic, and the number grows significantly when we consider the entire planet. The time spent in traffic is potentially dangerous, with more than 3,000 lives lost every day (2, 3) and most accidents due to human error (4). Autonomous vehicles have the potential to improve the quality and productivity of the time spent in cars, increase the safety and efficiency of the transportation system, and transform transportation into a utility available to anyone, anytime. This requires advances in many aspects of vehicle autonomy, ranging from vehicle design to control, perception, planning, coordination, and human interaction.

In this review, we focus on recent advances in planning and decision-making for autonomous vehicles, especially (*a*) how the vehicles decide where to go next, (*b*) how vehicles use the data provided by their sensors to make decisions with short and long time horizons, (*c*) how the interaction with other vehicles affects what to do, (*d*) how vehicles can learn how to drive from their history and from human driving, (*e*) how to ensure that the vehicle control and planning systems are correct and safe, and (*f*) how to ensure that multiple vehicles on the road at the same time coordinate and are managed to move people and packages to their destinations in the most effective way. Inspired by the possibility of a future where transportation becomes a utility, academic and industry communities have started to address the science and engineering of autonomy, and significant work has been directed toward these challenges. This article surveys recent results related to various aspects of decision-making and planning for autonomous vehicles.

The level of automation of an intelligent vehicle can vary from a human-operated vehicle to a completely self-driving, or autonomous, vehicle. SAE International outlines five levels of autonomy in their J3016 document (5). Up to level 2, the driver is required at all times; in levels 3 and 4, handovers from the vehicle to the driver in difficult situations are possible; and level 5 is reserved for vehicles that are fully autonomous under all circumstances. Traditionally, an incremental approach has been followed to introduce advancements in intelligent vehicles. These advancements increase the automation level in cars with systems that support the driver, e.g., to maintain a constant speed, follow a lane, or perform a car–driver handover (6).

Achieving the vision of fully capable automated vehicles will require overcoming many technical, legal, and social challenges (7). In this survey, we focus on technical approaches that aim to create a fully automated, or level 5, vehicle. The 2004–2007 Defense Advanced Research Projects Agency (DARPA)–sponsored competitions (8, 9) pushed research on automated driving to near-real-world conditions (10, 11). These methods were limited to relatively low speeds and clutter-free environments with a few moving obstacles. Impressive progress has been achieved in the last decade, yet recent works on self-driving vehicles still present limitations in the complexity of the environment and/or the speed of movement (12, 13).

Autonomous vehicles, which operate in complex dynamic environments, require methods that generalize to unpredictable situations and reason in a timely manner in order to reach human-level reliability and react safely even in complex urban situations. Informed decisions require accurate perception. Nonetheless, state-of-the-art computer vision systems cannot yet achieve error rates acceptable for autonomous navigation. Most recently, approaches combining decision-making, control, and perception have shown promising results. With the ever-increasing popularity of machine learning techniques and complex planning and decision-making methods, verification and guaranteed performance of the autonomous driving pipeline have become challenges still to be addressed.

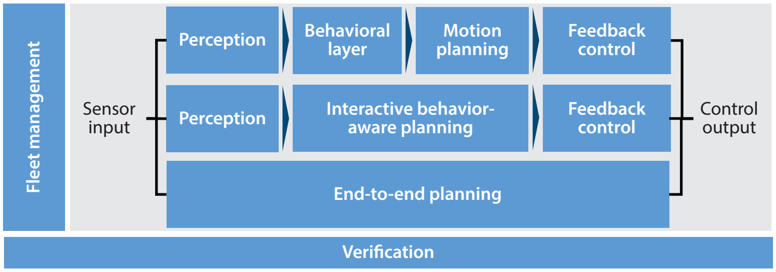

Figure 1

Schema of the planning and decision-making components described in this review: fleet management (Section 6); approaches for generating control commands from sensory data, namely traditional planning and control (Section 2), behavior-aware planning (Section 4), and end-to-end planning (Section 3); and verification of the methods for autonomous driving (Section 5).

In this review, we cover several aspects of planning and decision-making for autonomous vehicles. In particular, we distinguish between three distinct approaches: sequential planning, behavior-aware planning, and end-to-end planning (for a schematic overview, see **Figure 1**). The sequential approach utilizes advanced perception and decision-making methods to generate inputs for motion planning and control. After a brief overview of the state of the art in shared control of intelligent vehicles, these traditional methods for planning and control are described in Section 2. An alternative approach is that of integrated perception and planning, which includes learning-based end-to-end methods. These are described in Section 3, together with an overview of the state of the art in perception. The third approach is that of behavior-aware planning, where decision-making and planning are integrated into interactive planning. These methods are described in Section 4. In Section 5 we discuss how these methods could be verified or made safe by construction, and in Section 6 we describe methods for managing fleets of autonomous vehicles to provide mobility on demand. Finally, Section 7 concludes the article and provides several directions for future research.

2. MOTION PLANNING AND CONTROL

We first review traditional methods for vehicle control and motion planning in intelligent vehicles. We briefly introduce methods for parallel autonomy, where a human is still in control of the vehicle, and then focus on autonomous vehicles. At the end of this section, we provide an overview of several current challenges in decision-making and planning, which are then discussed in detail in the subsequent sections.

2.1. Vehicle Dynamics and Control

At relatively low speeds, a kinematic model of the car can be employed for control. Given a reference path, proportional–integral–derivative (PID) control, feedback linearization (14), or model predictive control can then be used to track it. However, operating at high speeds or performing aggressive maneuvers requires employing the full dynamic model of the vehicle, including tire forces (15–17). Nonlinear control (18), model predictive control (19), or feedback–feedforward control (20) stabilizes the behavior of the vehicle while tracking the specified path. Good tracking performance has been achieved with these vehicle models and controllers, even for autonomous racing.

These control methods rely on a model of the vehicle that needs to be identified. Both optimization-based and learning-based techniques for system identification exist (21). The chosen technique will depend on the amount and type of data available, the knowledge about the system dynamics, and the control method to be employed. Since the conditions of the road and the vehicle will vary with time, online model identification and lifelong system identification (22) will improve the performance of autonomous vehicles. Tools from machine learning show great potential to create models from the large amounts of data collected.

2.2. Parallel Autonomy

There are three types of collaborative autonomy: (*a*) series autonomy, in which the human orders the vehicle to execute a function, which is similar to most self-driving approaches to date; (*b*) interleaved autonomy, in which the human driver and the autonomous system take turns operating the vehicle; and (*c*) parallel autonomy (also referred to as shared control), in which the autonomous system functions as a guardian angel in the background to ensure safety while the human driver is operating the vehicle. Whether drivers are distracted or are simply overwhelmed by the difficulty of driving in challenging scenarios, a parallel autonomy framework offers additional safety. Many approaches for parallel autonomy have been proposed. In the following, we provide a brief overview of the field.

The most intuitive way of merging the human input with the output of a safety system is by linear combination of the two, as shown by Anderson et al. (23), who proposed threat measures based on the dynamic limitations of the vehicle. There, the human input was combined with a computed trajectory based on the severity of the threat. For example, shared control could be achieved via haptic feedback (24).

An alternative to input mixing is to directly incorporate the human inputs into an optimization framework in a minimally invasive manner. The objective is to minimize the deviation of the autonomous system's plan from the driver's intent. In its most basic form, the driver's intent is given by the current steering and acceleration inputs. A convex constrained optimization was employed by Alonso-Mora et al. (25) to compute safe inputs for shared control. However, the method was limited to a single-step look-ahead. A typical assumption for intelligent vehicles has been to consider the velocity of the vehicle as given and optimize only over the steering angle, thus rendering the optimization problem tractable. For example, Shia et al. (26) minimized the difference in steering wheel angle from the human predicted control input, which is necessary to achieve safe trajectories. Alternatively, Erlien et al. (27) defined vehicle stability and environmental envelopes to supply safe steering commands in a discretized environment, considering the vehicle speed to be constant and solving a receding-horizon convex optimization. Thanks to advances in fast nonlinear optimizers, it is now possible to optimize simultaneously over steering angle and velocity or throttle input (28, 28a) to achieve minimal intervention.

2.3. Motion Planning for Autonomous Vehicles

Two recent reviews (29, 30) provide a clear overview of the state of the art in motion planning for autonomous cars. In short, most traditional methods to compute safe trajectories for autonomous vehicles are based on one of three lines of thought. The first is input space discretization with collision checking, such as lattice planners (e.g., 31, 32) or road-aligned primitives (e.g., 33), whose main advantage is their simplicity and effectiveness, especially in highway scenarios. The second is randomized planning, such as rapidly exploring random trees (RRT) (e.g., 34, 35), whose main advantage is the probabilistic exploration of large state spaces, albeit at a high computational cost. The third is constrained optimization and receding-horizon control (e.g., 19, 36), which have been applied mostly to path following but now can also compute collision-free trajectories to avoid other traffic participants, as shown by Schwarting et al. (28), who formulated a nonlinear model predictive controller and employed it to safely navigate an intelligent vehicle. This has been possible thanks to recent advances in solvers for nonlinear constrained optimization. The main advantage of constrained optimization is the smoothness of trajectories and direct encoding of the vehicle model in the trajectory planning. Unfortunately, if not convex, constrained optimization converges only to a locally optimal trajectory for the vehicle.

Like human-driven cars, autonomous vehicles will also be subject to a large set of rules. These rules impose constraints on the motion planner, which should always be satisfied. However, under some circumstances (e.g., overtaking an illegally parked vehicle), they need to be violated. In this case, computing a trajectory that maximizes visibility (37) may help reduce risk, but, in general, the question of which rules shall be violated arises. If traffic rules are encoded in the cost function, traditional motion planning methods can be employed to find the path or trajectory of lowest cost. For example, Kuwata et al. (38) computed a cost map of the drivable space of the car and employed the RRT approach to find the path with the lowest cost. An alternative is to specify the rules as logic functions and utilize automatic control synthesis. For a discrete model of a robotic system and to reach a goal state, Tumova et al. (39) described a method to synthesize the motion that violates only the lowest-priority rules for the shortest amount of time. Although promising, challenges of automatic control synthesis still include their application to nondeterministic systems and environments, as well as continuous dynamic models, which is the case for autonomous vehicles. Similarly, Vasile et al. (40) considered the problem of minimum constraint violation in the context of integrated motion planning and routing in a road network. They utilized syntactically co-safe linear temporal logic (scLTL) formulae to specify the desired behavior of the vehicle and employed an RRT*-based motion planner to obtain a provably minimum-violation trajectory for a single-vehicle and single-trip scenario. Minimum-violation routing in the contexts of fleet management and vehicle sharing remains an open problem, which must be addressed to provide efficient transportation with minimal delays. In Section 6, we look at the fleet management and ride-sharing problem in more detail.

Most of the methods in this section consider a prediction over future trajectories of other traffic participants to be known. Yet real traffic scenarios include complex interactions among various road users. Handling complex clutter and modeling the interactions with other road users is necessary, and this remains an unsolved problem for autonomous driving. In Section 4, we look at this challenge in more detail. But first, in the next section we provide an overview of the state of the art in perception and end-to-end planning, which relies on machine learning. Finally, verification of the correctness and safety of the motion-planning methods is required to achieve broad applicability. We discuss this challenge in more detail in Section 5.

3. INTEGRATED PERCEPTION AND PLANNING

While the methods described in the previous section abstract perception away from planning, perception is of utmost importance for autonomous vehicles. In this section, we provide a brief overview of the state of the art in perception. This is followed by a description of end-to-end methods for integrated perception and planning, which generate a control input for the vehicle directly from sensory information and typically rely on machine learning.

3.1. From Classical Perception to Current Challenges in Neural Network–Based Perception Systems

A recent survey (41) includes both historical and the current state-of-the-art literature on several specific topics, including recognition, reconstruction, motion estimation, tracking, scene understanding, and end-to-end learning on several benchmarking data sets, including the KITTI (42), ISPRS (International Society for Photogrammetry and Remote Sensing), MOT (Multiple Object Tracking), and Cityscapes (43) data sets.

Classical perception systems extract information in the form of manually designed features from raw sensory data. The most notable examples are SIFT (Scale-Invariant Feature Transform) (44, 45), BRISK (Binary Robust Invariant Scalable Keypoints) (46), SURF (Speeded Up Robust Features) (47, 48), and ORB (Oriented FAST and Rotated BRIEF) (49, 50). Approaches building on hand-designed feature generators are constrained by the adaptivity to generic environments. By tracking these features, one can localize, estimate odometry, and map the environment in a simultaneous manner [simultaneous localization and mapping (SLAM)], which has become popular in the robotics community. While the current leaders in the KITTI Visual Odometry benchmark are based on lidar or a combination of vision and lidar (51), fast and lightweight approaches based purely on vision, such as ORB-SLAM2 (50), SVO (Semidirect Visual Odometry) 2.0 (52), and LSD-SLAM (Large-Scale Direct Monocular SLAM) (53), have matured to be able to compete. The high costs of lidar sensors are a strong driver for commercial and academic research and development in vision-based perception. While this may change owing to the development and availability of solid-state lidar, high-resolution radar, or other cheap depth sensors, a combination of multiple sensors with overlapping capabilities will most likely persist to achieve redundancy and increased safety.

While it may seem desirable to map the world in its finest detail to ease localization and planning in a predefined map, including pixel-perfect annotations of lane markings and traffic signs, this comes with several significant disadvantages. It may not be possible to update maps to reflect changes in the environment quickly enough. Highly detailed maps are expensive to create, maintain, and transfer, since updates need to be constantly fed into and distributed by the system. Therefore, it seems advantageous to keep only a light map containing condensed and semantic information; all other information, such as position inside a lane, traffic lights and signs, cars, and pedestrian detections, should be realized on the fly.

We refer the reader to an article by Bar Hillel et al. (54) for a survey on road and lane detection. Object detections are typically done by a bounding-box detection, maximizing the likelihood of detecting an object inside the box, or by semantic segmentation, classifying each pixel in the image space. For both tasks, deep neural network architectures have become dominantly successful. The current state of the art for object recognition may be found in the corresponding benchmarks, such as the ImageNet Large Scale Visual Recognition Challenge (55). In general, real-time-capable systems such as Faster R-CNN (Faster Regional Convolutional Neural Network) (56) exist.

By contrast, accurate semantic segmentation on high-resolution images in real time poses a greater challenge. State-of-the-art decoder–encoder network architectures, such as ResNet38 (57) and PSPNet (Pyramid Scene Parsing Network) (58), achieve more than 80% mIoU (mean intersection over union) in the Cityscapes data set (43) but take multiple seconds to propagate on high-resolution images, since they require a large number of floating-point operations. More recently, ENet (Efficient Neural Network) (59) achieved a 13-ms runtime on $1{,}024 \times 2{,}048$–pixel images with 58% mIoU on the Cityscapes data set (43), while ICNet (Image Cascade Network) (60) achieved 70% mIoU at 33 ms. ICNet incorporates multiresolution branches under proper label guidance to combine low-resolution layers (from which it learns the representation and extracts the most semantic information) and higher-resolution layers while simultaneously preserving details.

Deep neural network architectures rely on large amounts of data to generalize well enough to new environments and achieve sufficient variance reduction. Expensively manually labeled real-world data sets, such as the Cityscapes data set (43) for benchmarking semantic urban scene understanding, may contain only a limited amount of data. Artificial data from simulation, such as the SYNTHIA data set (61), which contains images for semantic segmentation of urban scenes, attempt to overcome this limitation. Johnson-Roberson et al. (62) offered a discussion of whether virtual worlds may replace human-generated annotations for real-world tasks. They compared training on an artificial data set of 200,000 images generated from simulation, based on the computer game *Grand Theft Auto V*, with training on the Cityscapes data set and evaluated the resulting networks on the KITTI data set (42) for vehicle detection. The network trained only on simulated car images significantly outperformed the one trained on real imagery (Cityscapes) on labels of all difficulties. Similarly, but for semantic segmentation, Richter et al. (63) created a data set from *Grand Theft Auto V*. Experiments on semantic segmentation data sets show that using the acquired data to supplement real-world images significantly increases accuracy and that the acquired data can reduce the amount of hand-labeled real-world data: Models trained with game data and just 1/3 of the real-world training set outperformed models trained on the complete real-world training set. Nonetheless, both approaches result in increased data set bias, which can also be found in real-world data sets (64).

A large issue of neural network–based perception systems is the insufficient feedback of uncertainty. Bayesian deep learning forms the intersection between deep learning and Bayesian probability theory, offering principled uncertainty estimates within deep architectures. The network's model uncertainty may be estimated with Monte Carlo dropout sampling (65), by propagating the given inputs through the network multiple times with different dropout weights. Evaluating the resulting statistics gives an estimate of the model uncertainty. As suggested by McAllister et al. (66), estimating and propagating uncertainty from every component throughout the entire system pipeline using a principled Bayesian framework would enable the autonomous vehicle to cope appropriately with high uncertainty.

The outputs of these low-level perception components are usually processed by a fusion component to generate a representation of the vehicle's environment (13). This environment model is then used by a further component to plan and control the vehicle's behavior. In the next section, we investigate how merging perception and planning may achieve closer coupling of sensory information and actuation.

3.2. End-to-End Planning

In conventional autonomous driving frameworks (13), functionalities are encapsulated with clear observable interfaces between modules. This may also be referred to as mediated perception (67), where objects of interest are detected and fused into a scene description, and driving commands are then computed.

Instead of keeping perception and planning modules separate, an alternative framework is to train certain parts of the perception module to incorporate partial tasks from the planning module. Caltagirone et al. (68) generated driving paths by integrating lidar point clouds, GPS–inertial measurement unit (IMU) information, and Google navigation information. The system is based on a fully convolutional neural network that jointly learns to carry out perception and path generation in the ground plane from real-world driving sequences. The method works in a weakly supervised manner, since driving paths can be labeled automatically from past driving data. Similarly, a semantic segmentation network may be employed to generate path proposals in the camera image space (69). Previously driven paths and obstacles detected from a lidar scanner are labeled and projected into the image space in an automated fashion. During deployment, only a camera image is needed to classify path proposals and obstacles. A major benefit of both approaches is the possibility of generating vast amounts of labeled data in an automated fashion without expensive manual label generation. Consequently, both systems may execute the path-planning function in a conventional planning pipeline.

Going a step further, one can learn the entire task of lane and road following without manual decomposition into road- or lane-marking detection, semantic abstraction, path planning, and control. ALVINN (Autonomous Land Vehicle in a Neural Network) (70) pioneered end-to-end driving in 1989 by teaching a neural network to output steering angles from camera images to keep the vehicle driving on the road. Chen et al. (67) referred to this as the behavior reflex approach, and by 2006, it was already possible to learn to avoid off-road obstacles from raw stereo-camera inputs (71). Since then, owing to the rise of GPU-computing capabilities for efficient learning of convolutional neural networks, common networks have become deeper and now contain many more parameters, increasing the overall performance of end-to-end driving. Researchers at NVIDIA (72) trained a deep convolutional neural network to map raw images from a front-facing camera directly to steering commands and were able to handle challenging scenarios such as driving on a gravel road, passing through roadwork, and driving during the night in poorly lit environments. During training, random shifts and rotations are applied to the original input image and virtual human interventions are simulated to artificially increase the number of training samples that require corrective control actions. By observing which regions of the input image contributed most to the output of the network (i.e., the salient objects), Bojarski et al. (73) showed that the network was capable of learning features resembling lane markings, road boundaries, and shapes of other vehicles, in an effort to explain the resulting behavior. Gurghian et al. (74) allowed for a close-up and uncluttered view of the lane by using two laterally mounted down-facing cameras and estimated the position inside the lane in an end-to-end fashion. The output of the lateral in-lane position may now be used to control the vehicle. Similarly, Chen et al. (67) trained a network to output affordance indicators, essentially features for the position and orientation inside the lane and other measures relative to other vehicles. Based on this simpler representation, a deterministic controller subsequently computes speed and steering commands.

Xu et al. (75) used a large-scale driving video data set to train an end-to-end fully convolutional long short-term memory network to predict both multimodal discrete behaviors (such as straight, stop, left turn, and right turn) on a task-based level and continuous driving behaviors (such as steering wheel angle control). The architecture for time-series prediction essentially fuses a long short-term memory temporal encoder with a fully convolutional visual encoder. Semantic segmentation as a side task further improves the model, following the privileged learning paradigm. In the same work, a large-scale data set of crowd-sourced driving from 21,808 unlabeled dashboard camera videos of different behaviors was automatically labeled and used for training.

DAgger (Dataset Aggregation) works in a setting where the reward is given only implicitly and improves upon supervised learning by letting a primary policy collect training examples while simultaneously running a reference policy. This dramatically improves the performance of a primary policy. SafeDAgger (76) is a query-efficient extension to DAgger (77). To achieve query efficiency in SafeDAgger, a safety classifier is introduced to predict the error made by a primary policy without querying a reference policy.

End-to-end motion planning has also been applied to robotics—for example, to learn a navigation policy in simulation from an expert operator, with a 2-D laser range finder and relative goal position as inputs (78). It is then feasible to transfer the knowledge gained from training to unseen real-world environments to perform target-oriented navigation and collision avoidance. Socially aware collision avoidance with deep reinforcement learning was introduced to explain and induce socially aware behaviors capable of learning directly from multiagent scenarios by developing a symmetrical neural network structure (79).

Robots that use learned perceptual models in the real world must be able to safely handle cases where they are forced to make decisions in scenarios that are unlike any of their training examples. Recent ensemble, bootstrap, and Monte Carlo dropout methods for quantifying neural network uncertainty (Bayesian neural networks) may not be able to efficiently provide accurate uncertainty estimates when queried with inputs that are very different from their training data. Therefore, an autoencoder may be trained to recognize when to detect novelties in the input data (80) and revert from an end-to-end approach to a safe non-learning-based behavior, such as the execution of conventional motion primitives.

Another line of research learns driving behavior in simulation, making it suitable for reinforcement learning because it is possible to observe failure cases during learning in a safe environment. The approaches presented above inherit only normal driving behavior, which may indicate that they cannot operate well in rare corner cases, such as crashes. In the best case, reinforcement learning may actively seek these difficult cases during exploration. Additionally, ground truth perception information is available in simulation, easing the definition and computation of a reward function. Wolf et al. (81) presented an approach for learning to steer a vehicle in a simulation environment using a Deep Q-Network. Nonetheless, the action space is discrete, allowing only for coarse steering wheel adjustments. They found that, when benchmarking for distance from the lane center, overall performance can be increased by adding other terms, such as the angle deviation of the vehicle from the center line. The gap between simulation and real-world data could be closed (82) by first segmenting the virtual image from the simulator with a segmentation network and then translating it into a realistic-looking image employing a generative network. The generative network is trained to create seemingly real-looking images from segmentations. To operate over continuous action spaces, Lillicrap et al. (83) proposed an actor–critic and model-free algorithm that is based on the deterministic policy gradient and relies on deep reinforcement learning. The algorithm is able to learn a policy to remain on the track in a simulated car-driving environment.

4. BEHAVIOR-AWARE MOTION PLANNING

Most of the methods in Section 2 expect a prediction over the future trajectories of other traffic participants in order to avoid collisions, but real traffic scenarios involve complex interactions among various road users. Handling complex clutter and modeling interactions with other road users are necessary to provide safety. In this section, we investigate this open challenge.

In the DARPA Urban Challenge, multiple solutions for tactical planning were proposed, although they were specifically tailored to the challenge's needs. Most approaches (e.g., 10, 11, 84) use a state machine to switch between predefined behaviors. These rule-based approaches lack the ability to generalize to unknown situations and deal with uncertainties.

Automated driving with humanlike driving behavior requires interactive and cooperative decision-making. Other motorists' intentions need to be deduced and integrated into a planning framework that allows for reasonable cooperative decision-making without the need of intervehicle communication. While autonomous vehicles need to be able to deduce the intentions of other human traffic participants, they also need to enable others to reasonably infer the autonomous vehicle's intention. This results in interdependencies and interactions based on the seen and shown behavior without the need for explicit communication.

We first present work in the area of cooperative and socially compliant behavior planning and then expand in a later section to more general interactive planning, including interactiveness with other agents and the environment by modeling, or actively reducing, uncertainty due to occlusions and incomplete sensor information. We then discuss learning-based approaches.

4.1. Cooperation and Interaction

Socially compliant driving, including cooperation and interactivity, not only are important to create congruent behavior among real human drivers but also are vital for safe navigation in cluttered, dynamic, and uncertain environments. Since an agent's actions are interdependent on all other agents' actions, an uncertainty explosion in future states arises and results in the freezing-robot problem (85) discussed in the robotics community. The robot comes to a complete stop because all possible actions become unacceptably unsafe. If the robot does not come to a complete stop, it will choose to follow highly evasive or arbitrary paths through the crowd that are often not only suboptimal but potentially dangerous.

There are now essentially three ways to tackle the issue of exploding uncertainty:

- Find a better description of the dynamics of the environment, including dynamic obstacles, as in partially closed-loop receding-horizon control (86), by modeling the anticipated future information to reduce the uncertainty that is associated with future belief states. However, even under perfect individual prediction and perfect knowledge of all agents' trajectories, the freezing-robot problem cannot always be prevented (85).
- Model cooperation based on a conditional formulation that models how the agents react to the robot's actions (as in 87). One problem is that modeling the reactions to the robot's behavior indirectly assumes the ability to fully control all other agents. Intuitively, assuming full control over all agents may lead to aggressive and potentially dangerous behavior owing to overconfidence in the behavior model.
- Model cooperation via joint distributions, i.e., essentially modeling the robot as one of the other agents. Examples are joint probability distributions (85) and joint cost distributions (88).

An agent's behavior can be defined as cooperative if joint utility is knowingly and willingly increased in comparison with a reference utility (89). For the purpose of comparability, we consider approaches that, in some form, increase joint utility by incorporating the goals of other agents as cooperative. In this context, cooperative motion planning may also be referred to as a goal for socially compliant motion planning. A survey of cooperative planning (90) distinguishes cooperative driving behavior into the two dimensions of intervehicle communication and cooperation in the sense of collaboration. In this review, we focus on approaches that do not rely on communication with other vehicles or infrastructure.

4.2. Game-Theoretic Approaches

It is a common pattern to model other vehicles' behavior as expected utility maximizing—i.e., an agent is expected to execute the most beneficial controls (87). Therefore, a reward or utility function needs to be known or learned. This can be done in a similar manner for probabilistic approaches, where instead of optimizing for lowest cost, the vehicle's controls are expected to follow the rule of maximum likelihood or maximum a posteriori. The actions are typically rolled out and scored over a fixed time horizon, resulting in a receding-horizon planner.

Another distinction between approaches is whether this optimization is done for a joint cost or distribution or in a two-player game, where the autonomous vehicle first computes an action and then models the other vehicle to react in a way that maximizes its own expected reward. The emergent behavior can be highly interactive instead of reactive, because the autonomous vehicle will optimize to maximize its own reward, which is dependent on the other vehicle's actions. The latter process results in an assumption of indirect control over the other vehicle.

While modeling interactions is an intriguing problem in itself, dealing with the increased complexity is another challenge. Since all agents' actions are affected and equally affect other agents' actions, the number of interactions (and therefore the planning complexity) grows exponentially with the number of agents. The simplest approach is to discretize the action space by motion primitives and to exhaustively search through all possible options (89). Naturally, there are more efficient methods of exploring the optimization space. In the deterministic case, one can cover the decision-making process, often phrased in a game-theoretic setting (91), in a tree-type structure and apply a search over the tree. The tree, usually discretized by action time, consists of discrete actions that each agent can choose to execute at each stage of the tree. Since each agent's reward depends not only on its own reward and actions but also on all other agents' actions at the previous stages, the tree grows exponentially with the number of agents. To achieve faster optimization for an optimal (or approximately optimal) solution, other tree search algorithms, such as Monte Carlo tree search (92), may be applied. To reduce computational complexity, Schwarting & Pascheka (93) assumed that the following vehicles' actions are dominated by their predecessors and used this assumption to formulate a recursive conflict-resolution algorithm to achieve only quadratic complexity in the number of agents.

Li et al. (94) modeled the decision-making in autonomous driving as a Stackelberg game. The autonomous vehicle, the leader, chooses its actions to maximize its utility for the worst-case actions that following vehicles might choose. All other vehicles act similarly in a leader–follower chain. Therefore, not all vehicles' actions are interdependent with all other vehicles' actions, and the complexity grows only linearly with the number of agents, as compared with decision trees. The approach shows the feasibility of solving the decision-making game even for more than 30 vehicles in real time. Nonetheless, in a comparison of Stackelberg and decision trees for decision-making, decision trees outperformed the Stackelberg approach in both average speed and number of constraint violations.

4.3. Probabilistic Approaches

In a highway entry scenario involving an autonomous vehicle merging into moving traffic, Wei et al. (95) planned for two vehicles to execute a set of possible high-level policies in a Markov decision process. A search for the best policy is performed by forward simulating to find the most likely traffic scenario and then executing the corresponding policy from the set of available policies for the ego vehicle. Every policy is then scored against the ego vehicle's cost function, and the best policy is executed. The authors associated social behavior with a simple Bayes model: Other vehicles are more likely to yield if decelerating and less likely to yield if accelerating. No reciprocal reward-based model is employed.

Trajectories can also be sampled on a discretized manifold (96), similar to the work of Werling et al. (33), and the environment's reaction can be rolled out according to the intelligent driver model. As a time-continuous car-following model for the simulation of freeway and urban traffic, the intelligent driver model describes the dynamics of the longitudinal positions and velocities of single vehicles in a traffic flow on a micro level. The approach incorporates cooperative behavior by including other vehicles' efforts (acceleration) into a joint cost function and therefore achieves a certain level of cooperation. An additional constraint on the other vehicles' maximal acceleration is enforced. Hoermann et al. (97) used a particle filter to estimate the intelligent driver model's behavior parameters, corresponding to maximum acceleration, desired acceleration, desired velocity, minimum distance, and desired time gap. The resulting posterior density is used to probabilistically propagate the current state to receive probabilistic long-term predictions for autonomous vehicles in a longitudinal direction.

Dong et al. (98) used a probabilistic graphical model to describe the dependency among observed data and estimate other cars' intentions. The task of the probabilistic graphical model is to generate an intention estimation with maximum probability, given observed information.

In interacting Gaussian processes (85), each agent's trajectory is modeled via a Gaussian process. Individual Gaussian processes are coupled through an interaction potential that models cooperation between different agents' trajectories. Terms for affordance, for progress, and to penalize close distances to other agents can also be included in their joint cost function (88).

4.4. Partially Observable Markov Decision Processes

In the probabilistic case, the problem is often formulated as a partially observable Markov decision process (POMDP), where the intentions and replanning procedures of the other agents are not directly observable and are encoded in hidden variables. Publications in the POMDP community typically focus on solving POMDP models offline. In this context, offline means that the focus is typically to calculate the best possible action not for the current belief state but rather for every imaginable belief state. Hence, they provide a policy—prior to the execution—of the best action to execute in any possible situation. POMDP problems are PSPACE complete and thus computationally intractable for large state spaces. Even for relatively small POMDP problems, it takes several minutes to hours to calculate approximate offline solutions. By contrast, for decision making in traffic environments, decisions need to be updated frequently (e.g., every 100 ms). Since solving the most general POMDP is intractable in real-time applications, approximate POMDP solutions to simplified problem formulations are employed to avoid the complexity of computing a sophisticated, long-term policy.

A POMDP with the other vehicle's intentions as hidden variables can be employed as well (99). The proposed method simplifies the problem significantly by planning all vehicles' motions on preplanned paths, reducing the dimensionality of the state space of the given problem. The POMDP formulation readily incorporates trade-offs among exploration (the information-gathering process) and exploitation (the progress toward a goal or reward). Nonetheless, the interaction model simply consists of a constant braking action triggered if a time to collision falls below a threshold. A POMDP integrating the road context and the motion intention of another vehicle in an urban road scenario was solved by Liu et al. (100). A reference vehicle behavior corresponding to the road context is defined, and the other vehicle's reaction is inferred by observing the deviation from the reference behavior. A discretization of the other vehicle's intentions (i.e., a hidden variable) allows the approach to infer other vehicles' intentions, such as giving way or acting aggressively.

It is also possible to plan, without interactions, over specific regions of interest (101) instead of the whole set of other vehicles and only for the current belief state. This is typically done by a look-ahead search in the belief state space to explore only those belief states that are actually reachable from the state right now.

Oftentimes, a large amount of domain knowledge can be incorporated into the action selection process to simplify the decision-making; planning horizons are relatively short, since predictions are accurate for less than 10 s. In both highway and city driving, the number of distinct actions, although they may vary during execution, remains relatively small. Only limited planning accuracy is needed in the far future. Planning on abstractions rather than detailed trajectories can lower planning complexity significantly. Ulbrich & Maurer (101) applied a tree-based policy evaluation that incorporated the above-described domain knowledge. The problem setting of growing complexity is then again similar to the deterministic setting. Likewise, Galceran et al. (102) suggested a custom POMDP solver that forward propagates multiple hand-defined policies, including hyperparameters, and computes the deterministic closed-loop feedback on the autonomous vehicle's policies. The main approximation is in reducing the decision to a limited set of policies and performing evaluations with a single set of policy assignments for each sample. Alternatively, Zhou et al. (102a) proposed a real-time method for joint multipolicy behavior estimation and receding-horizon trajectory planning in urban environments. The authors employed a coupled POMDP to estimate the future trajectory of the interacting traffic participants and a chance-constrained nonlinear MPC planner (extension of Reference 28) to compute safe trajectories.

Finding a suitable symbolic representation for the POMDP is difficult, as it heavily depends on the specific task and situation. The usual approach is to use an equidistant discretization of the continuous space. On the one hand, such a discretization is often too coarse and cannot represent enough detail to find a solution to the problem. On the other hand, it encodes information redundantly where high precision is not needed. Brechtel et al. (103) presented a continuous POMDP with a focus on balancing exploration and exploitation in the scenario of occlusions and incomplete perception. The continuous POMDP is solved by incremental learning of an efficient space representation during value iteration. While reasoning about potentially hidden objects and observation uncertainty, they also consider the interactions of road users.

A different approach, instead of solving the POMDP in a conventional way or by domain knowledge and specific simplifications, is to employ nonparametric reinforcement learning (as in 104), to immediately receive an approximately optimal policy without optimization. However, generalization to arbitrary environments remains a challenge.

4.5. Learning-Based Approaches

In the previous sections, we have focused on the frameworks and models for interactions among human-driven vehicles. We now turn to data-driven approaches. We exclude approaches related to end-to-end driving, which were already presented in Section 3.2, and continue focusing on behavior-aware motion planning.

Typical approaches decouple decision-making and planning. For instance, Vallon et al. (105) trained a support vector machine for lane-change decision-making with features composed of relative position and relative velocity. If a lane-change desire is triggered, a lane-change reference trajectory is executed by a model predictive controller with the objective from minimal deviation to the reference subject to a set of safety constraints.

Gaussian mixture models parameterized by neural networks with features based on the ego vehicle's and the surrounding vehicles' states, past actions, and specifications and the road geometry were trained in Lenz et al. (106) to predict the motion of a group of vehicles in a highway setting. Fully connected layers were able to outperform recurrent neural networks as well as other able classical models, such as the intelligent driver model.

An alternative is to employ a nonparametric prediction architecture (107). A sample generation module consisting of a conditional variational autoencoder was able to learn a sampling model that, given observations of past trajectories, produces a diverse set of prediction hypotheses to capture the multimodality of the space of plausible futures. An inverse optimal control–based ranking module determines the most likely hypothesis while incorporating scene context and interactions. Interestingly, this work shows similarities to the sampling hypothesis and subsequent scoring and refinement used by numerical POMDP solvers.

Inverse reinforcement learning (IRL) is a prominent framework. In the literature, IRL is also often referred to as inverse optimal control. In all of these cases, an unknown reward function is learned from expert demonstrations. In classical approaches, the cost function consists of a weighted sum of hand-designed features $\Phi(s) = \sum_{k=1}^{p} w_k \phi_k(s)$, where the weights w_k are to be learned.

As mentioned above, interactions can be modeled by indirect control over the other vehicle (87), in the manner of an underactuated system. The proposed method learns the reward function via feature-based IRL from expert demonstrations. Manually designed features (i.e., cost terms) incorporate objectives of staying inside lanes, collision avoidance, a measure for progress, and control effort costs. Other vehicles' behaviors follow from a two-player game where the other vehicle maximizes its own reward in response to a given control trajectory of the autonomous vehicle. As such, the human driver is assumed to act egoistically. This approach is able to leverage effects of the autonomous vehicle's behavior on human actions. Emergent behavior includes induced lane changes and changes in velocity at intersections and highway segments. An extension (108) gathers information about the internal state of another vehicle's driver by including the information gain over a belief state in the objective function, effectively reducing the entropy. The belief state encodes the affiliation with one of two discrete cost functions modeling the driver behavior, e.g., attentive versus distracted. In contrast to related POMDP formulations, the exploration–exploitation trade-off is not addressed yet and is encoded only by a linear combination of objectives in the reward function. By contrast, the weights of the reward function can also be found by having a human driver choose a preferred trajectory iteratively from a set of two candidate trajectories (109). This allows the vehicle to learn the reward function without a set of expert trajectories and predefined labels.

In a further step toward communicating robot objective functions to people, Huang et al. (110) recognized that, unlike robots, humans will not be exact in their IRL inference. They introduced a collection of approximate-inference models and, in a user study, showed increased performance in comparison with an exact-inference model.

An exemplary implementation of learning different driving styles in a highway simulation showed the potential of Markov decision processes with an unknown reward function (111). Abbeel et al. (112) demonstrated an improved version of the algorithm and its performance by generating humanlike trajectories in parking lots, with only a few demonstrations required during learning.

Ziebart et al. (113) applied the principle of maximum-entropy IRL, which is a natural choice to avoid overfitting since the maximum-entropy distribution shows the least commitment to the data. Maximum-entropy IRL has been popular for learning cost functions in robotics and autonomous driving. Kuderer et al. (114), Kretzschmar et al. (88), and Pfeiffer et al. (115) described learning socially compliant motion planning and human behavior. Herman et al. (116) presented an approach for priority adaptive navigation, where a robot must choose navigation models of different social acceptabilities based on task constraints. The behavioral models are learned by maximum-entropy IRL from demonstrations of different social acceptabilities. A similar variant, maximum margin planning (117), was applied to navigate a robot in complex unstructured terrain (118) and to learn autonomous driving styles and maneuvers (119).

Continuous inverse optimal control with locally optimal examples (120) may be used to handle continuous states and actions and the fact that expert demonstrations may be noisy and possibly locally optimal. Levine & Koltun (120) showed the ability to learn aggressive and evasive driving styles from demonstrations based on features consisting of speed, deviation from lane centers, and Gaussian distributions covering the front, back, and sides of the other cars on the road. Similarly, Sadigh et al. (87) demonstrated the ability to learn human driver rewards as a model for human behavior.

Majumdar et al. (121) devised a framework for risk-sensitive IRL to be able to take an expert's risk sensitivity explicitly into account. This framework was capable of capturing ranges of different risk preferences, from risk neutral to worst case. A linear programming–based algorithm was used to infer an expert's hidden risk metric.

The maximum-entropy deep IRL framework (122) exploits the expressive capacity of deep fully convolutional neural networks to represent the cost model underlying driving behaviors. In general, deep fully convolutional neural networks, as robust, flexible, high-capacity function approximators, are able to model the complex relationship between sensory input and reward structure very well. Additionally, thanks to convolutional operators, they are able to capture spatial correlations in the data. Wulfmeier et al. (123) were able to learn an end-to-end mapping from raw input data to cost map from more than 25,000 demonstrations over 120 km of driving.

Lastly, Kuefler et al. (124) demonstrated the effectiveness of generative adversarial imitation learning (125), extended to the optimization of recurrent policies. As discussed above, one approach to learning policies from expert demonstrations is to recover the expert's cost function with IRL and then extract a policy from that cost function with reinforcement learning. Since this direct procedure is usually slow, generative adversarial imitation learning poses a framework to extract policies directly from data. The approach reproduces emergent behavior of human drivers, such as congruent lane-change behavior, while maintaining validity over long time horizons.

5. VERIFICATION AND SYNTHESIS

Recent studies (126) have indicated that the minimal requirement to demonstrate safety for an autonomous car is hundreds of millions of miles, taking possibly tens of years to complete. While simulation and case-based testing are routinely employed to check the performance of autonomy methods, they do not provide sufficient guarantees. This is especially true for safety-critical domains such as autonomous driving, where unsafe events are rare and hard to characterize. To meet these proof-of-safety demands, we need frameworks that provide analytical proofs of safety, rather than checking a finite set of concrete traffic situations in simulation.

Given a model of the system and environment, safe controllers can be produced by model-based correct-by-construction synthesis, meaning that trajectories of the closed-loop systems provably meet the specification. Synthesis approaches typically rely on specifications given in linear temporal logic and have been developed for low-complexity tasks such as adaptive cruise control (127) and control of signalized vehicular networks (128). Yet controller synthesis is currently limited in scope and deployment owing to its very large computational cost.

An alternative to correct-by-construction synthesis is formal verification, which typically has a smaller computational footprint. Model checking is a widely used technique for formal verification of distributed systems. It examines the complete reachable state space of a model in order to determine whether the system satisfies its requirements or desired properties. This approach was applied to formally verify the state consistency between different software modules of the autonomous vehicle developed by the California Institute of Technology for the 2007 DARPA Urban Challenge (129) and for verification of adaptive cruise control (130). Online verification can be achieved with reachability analysis (131) when performing conservative linearization, using zonotopes as a set representation and querying from a database of specific emergency maneuvers. Online verification of general maneuvers would require a probabilistic representation of other traffic participants and scenarios, which is still computationally challenging because of the high complexity. The drivability of planned motions can also be checked for, and guaranteed, via reachability analysis (132). Instead of verifying car controllers online, an alternative is to build a library of local and verified road models, such as intersections and road segments, that are composed together to certify safety over networks (133). Such an approach can work well to verify controllers and networks (e.g., for urban planning) but do not yet account for all uncertainties in the behavior of traffic participants.

Formal synthesis and verification have been extensively studied in the field of control, and they are prominent tools to guarantee the safety of autonomous vehicles, despite their high computational cost and associated limitations. In parallel, there has been an ever-increasing popularity of artificial intelligence techniques, such as neural networks, which raise questions about safety since their output response is not well known, especially outside of the training data regime. Although traditional tools, such as short-term memory solvers, could be extended for verification of deep neural networks (134), additional challenges exist for verification of artificial intelligence. Seshia et al. (135) identified five challenges: modeling of the complex environment, modeling of the system, formal specification of the desired properties of the system, scalability, and formal quantification of the requirements for training data.

6. FLEET MANAGEMENT

Ride-sharing services are transforming urban mobility. Also known as vehicle-pooling options, these systems allow several passengers to share a vehicle when traveling along similar routes. These companies currently rely on drivers to operate the vehicles, but there is a push in the industry toward autonomous self-driving vehicles. These fleets of autonomous vehicles are expected to provide safe, reliable, and affordable transportation. In this section, we discuss approaches for dynamic vehicle routing and passenger assignment.

Much of the fleet management literature for mobility-on-demand systems considers the case of ride sharing without pooling requests, focusing on fluid approximations (136), queuing-based formulations (137), and case studies in specific regions. With the growing interest and rapid developments in autonomous vehicles, there is also an increasing focus on autonomous mobility-on-demand systems (138). However, none of these works considered the ride-pooling problem of servicing multiple rides with a single trip. The ride-pooling problem is more related to the vehicle-routing problem and the dynamic pickup and delivery problem (139–141), where spatiotemporally distributed demand must be picked up and delivered within prespecified time windows. A major challenge when addressing this problem is the need to explore a very large decision space while computing solutions fast enough to provide users with the experience of real-time booking and service.

A study in New York City showed that up to 80% of the taxi trips in Manhattan could be shared by two riders with an increase in the travel time of a few minutes (142) and also showed the gains attainable by a "global oracle" with full knowledge of the future. These results were confirmed by Alonso-Mora et al. (143), who introduced an anytime-optimal method for request matching and dynamic vehicle routing in low- and high-capacity vehicles. The method, which consists of three steps—pruning of feasible trip combinations, assignment of trips to vehicles, and fleet rebalancing—showed that large-scale operation of vehicle fleets is possible.

These works, and their predecessors, have opened the way for several avenues of research, where artificial intelligence will play a stronger role (144). Powerful data-mining tools and readily available large data sets of public transit data (145) will allow us to build probabilistic models of future travel patterns and use them to better position the fleet of vehicles for future requests. These models will then be included in probabilistic and uncertainty-aware large-scale planning methods that in expectation guarantee a certain quality of service. A recent review of stochastic routing highlighted state-of-the-art works in this area (146). For instance, Alonso-Mora et al. (147) computed a historical probability distribution of future requests and included random samples in the dynamic routing of the vehicles and the passenger assignment, within the context of ride sharing.

With naive fleet management approaches and an ever-increasing number of vehicles and people, congestion could be a problem (148). Real-time traffic data will also be employed to achieve congestion-aware routing and navigate autonomous cars in a way that minimizes congestion. In this direction, Zhang et al. (149) described a constrained optimization method for congestion-aware routing in single-seat vehicles, and Levin (150) introduced a fluid-approximation approach that also accounts for vehicle sharing. Yet large-scale efficient routing in real time is still a challenge, especially in the context of ride sharing. Furthermore, taking into account the intentions of other autonomous vehicles is also important in the context of electric vehicles, which can only charge their batteries at a finite number of locations (151).

In the future, we will see multimodal transportation with a combination of various autonomous vehicles, such as taxis and buses. We believe that decoupling approaches (e.g., 143) are promising for large-scale vehicle routing. But for now, the challenge resides in scaling current techniques to city-sized problems involving millions of trips per day. Finally, regulation, privacy, and human supervision of large fleets are also avenues of future research where artificial intelligence will play an important role.

7. CONCLUSION

This review has provided an overview of current advances in planning and decision-making for autonomous vehicles. While the field has made tremendous progress over the last few years, many questions remain unanswered. The increased popularity of data-driven algorithms in both perception systems and planning systems requires a second wave of innovation; verifiability, safety, and explainability are key requirements to allow the transition from systems suitable for showcases toward production-ready autonomous vehicles in our everyday lives. Additionally, autonomous systems that operate in complex, dynamic, and interactive environments require artificial intelligence that generalizes to unpredictable situations and reasons in a timely manner about the interactions with many traffic participants. Autonomous systems still need to reach human-level reliability in decision-making, planning, and perception, and current detection and segmentation accuracies do not yet suffice in difficult conditions, such as inclement weather. Finally, autonomous vehicles will provide on-demand transportation potentially to anyone, anywhere, anytime. To achieve this vision, further advances are also required in large-scale fleet management with stochastic routing, online performance, and bounded quality of service. If we can overcome these challenges, autonomous vehicles will have a tremendously beneficial impact on our lives.

FUTURE ISSUES

1. Planning methods are needed that provide safe and system-compliant performance in complex cluttered environments while modeling the interaction with other traffic participants.
2. Planning and perception need to be closely integrated for direct propagation of uncertainty and features with safety guarantees.
3. Planning and control in inclement weather must be improved.
4. Machine learning approaches for planning and decision-making need to be developed, evaluated, and integrated.
5. The methods employed in autonomous vehicles will require verification and safety assessments.
6. Methods will need to be developed for large-scale fleet management with stochastic routing, online performance, and bounded quality of service.

DISCLOSURE STATEMENT

The authors are not aware of any affiliations, memberships, funding, or financial holdings that might be perceived as affecting the objectivity of this review.

ACKNOWLEDGMENTS

The Toyota Research Institute (TRI) provided funds to assist the authors with their research, but this article reflects solely the opinions and conclusions of its authors and not those of TRI or any other Toyota entity.

LITERATURE CITED

1. Fed. Highw. Adm. 2015. *U.S. driving increases for sixth straight year, new federal data show*. Press Release, Fed. Highw. Adm., US Dep. Transp., Washington, DC. **https://www.fhwa.dot.gov/pressroom/fhwa1711.cfm**
2. Assoc. Safe Intl. Road Travel (ASIRT). 2017. Home page. **http://www.asirt.org**
3. Natl. Saf. Counc. (NSC). 2015. *NSC motor vehicle fatality estimates 2012–2015*. Rep., Stat. Dep., NSC, Itasca, IL. **http://www.nsc.org/NewsDocuments/2016/mv-fatality-report-1215.pdf**
4. Natl. Highw. Traffic Saf. Adm. (NHTSA). 2015. *Traffic safety facts 2015*. Rep., NHTSA, US Dep. Transp., Washington, DC. **https://crashstats.nhtsa.dot.gov/Api/Public/ViewPublication/812384**
5. SAE Intl. 2016. *Taxonomy and definitions for terms related to driving automation systems for on-road motor vehicles*. Stand. J3016, SAE Intl., Warrendale, PA
6. Russell HEB, Harbott LK, Nisky I, Pan S, Okamura AM, Gerdes JC. 2016. Motor learning affects car-to-driver handover in automated vehicles. *Sci. Robot.* 1:eaah5682
7. Maurer M, Gerdes JC, Lenz B, Winner H. 2016. *Autonomous Driving: Technical, Legal and Social Aspects*. Berlin: Springer
8. Buehler M, Iagnemma K, Singh S. 2007. *The 2005 DARPA Grand Challenge: The Great Robot Race*. Berlin: Springer
9. Buehler M, Iagnemma K, Singh S. 2009. *The DARPA Urban Challenge: Autonomous Vehicles in City Traffic*. Berlin: Springer
10. Urmson C, Anhalt J, Bagnell D, Baker C, Bittner R, et al. 2008. Autonomous driving in urban environments: Boss and the Urban Challenge. *J. Field Robot.* 25:425–66
11. Leonard J, How J, Teller S, Berger M, Campbell S, et al. 2008. A perception-driven autonomous urban vehicle. *J. Field Robot.* 25:727–74
12. Furgale P, Schwesinger U, Rufli M, Derendarz W, Grimmett H, et al. 2013. Toward automated driving in cities using close-to-market sensors: an overview of the V-Charge Project. In *2013 IEEE Intelligent Vehicles Symposium (IV)*, pp. 809–16. New York: IEEE
13. Ulbrich S, Reschka A, Rieken J, Ernst S, Bagschik G, et al. 2017. Towards a functional system architecture for automated vehicles. arXiv:1703.08557
14. De Luca A, Oriolo G, Samson C. 1998. Feedback control of a nonholonomic car-like robot. In *Robot Motion Planning and Control*, ed. JP Laumond, pp. 171–253. Berlin: Springer
15. Gillespie TD. 1997. *Vehicle Dynamics*. Warrendale, PA: Soc. Automot. Eng.
16. Pacejka H. 2012. *Tire and Vehicle Dynamics*. Oxford, UK: Elsevier. 3rd ed.
17. Rajamani R. 2012. *Vehicle Dynamics and Control*. New York: Springer. 2nd ed.
18. Hoffmann GM, Tomlin CJ, Montemerlo M, Thrun S. 2007. Autonomous automobile trajectory tracking for off-road driving: controller design, experimental validation and racing. In *2007 American Control Conference*, pp. 2296–301. New York: IEEE
19. Falcone P, Borrelli F, Asgari J, Tseng HE, Hrovat D. 2007. Predictive active steering control for autonomous vehicle systems. *IEEE Trans. Control Syst. Technol.* 15:566–80
20. Kapania NR, Gerdes JC. 2015. Design of a feedback-feedforward steering controller for accurate path tracking and stability at the limits of handling. *Vehicle Syst. Dyn.* 53:1687–704
21. Nelles O. 2001. *Nonlinear System Identification: From Classical Approaches to Neural Networks and Fuzzy Models*. Berlin: Springer

22. Seegmiller N, Rogers-Marcovitz F, Miller G, Kelly A. 2013. Vehicle model identification by integrated prediction error minimization. *Int. J. Robot. Res.* 32:912–31
23. Anderson SJ, Karumanchi SB, Iagnemma K, Walker JM. 2013. The intelligent copilot: a constraint-based approach to shared-adaptive control of ground vehicles. *IEEE Intell. Transp. Syst. Mag.* 5:45–54
24. Abbink DA, Mulder M, Boer ER. 2011. Haptic shared control: smoothly shifting control authority? *Cogn. Technol. Work* 14:19–28
25. Alonso-Mora J, Gohl P, Watson S, Siegwart R, Beardsley P. 2014. Shared control of autonomous vehicles based on velocity space optimization. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1639–45. New York: IEEE
26. Shia VA, Gao Y, Vasudevan R, Campbell KD, Lin T, et al. 2014. Semiautonomous vehicular control using driver modeling. *IEEE Trans. Intell. Transp. Syst.* 15:2696–709
27. Erlien SM, Fujita S, Gerdes JC. 2016. Shared steering control using safe envelopes for obstacle avoidance and vehicle stability. *IEEE Trans. Intell. Transp. Syst.* 17:441–51
28. Schwarting W, Alonso-Mora J, Paull L, Karaman S, Rus D. 2017. Parallel autonomy in automated vehicles: safe motion generation with minimal intervention. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1928–35. New York: IEEE
28a. Schwarting W, Alonso-Mora J, Paull L, Karaman S, Rus D. 2018. Safe nonlinear trajectory generation for parallel autonomy with a dynamic vehicle model. *IEEE Trans. Intell. Transp. Syst.* In press. **https://doi.org/10.1109/TITS.2017.2771351**
29. Katrakazas C, Quddus M, Chen WH, Deka L. 2015. Real-time motion planning methods for autonomous on-road driving: state-of-the-art and future research directions. *Transp. Res. C* 60:416–42
30. Paden B, Cap M, Yong SZ, Yershov D, Frazzoli E. 2016. A survey of motion planning and control techniques for self-driving urban vehicles. *IEEE Trans. Intell. Veh.* 1:33–55
31. Ferguson D, Howard TM, Likhachev M. 2008. Motion planning in urban environments. *J. Field Robot.* 25:939–60
32. Pivtoraiko M, Knepper RA, Kelly A. 2009. Differentially constrained mobile robot motion planning in state lattices. *J. Field Robot.* 26:308–33
33. Werling M, Kammel S, Ziegler J, Gröll L. 2012. Optimal trajectories for time-critical street scenarios using discretized terminal manifolds. *Int. J. Robot. Res.* 31:346–59
34. LaValle SM, Kuffner JJ. 2001. Randomized kinodynamic planning. *Int. J. Robot. Res.* 20:378–400
35. Karaman S, Frazzoli E. 2011. Sampling-based algorithms for optimal motion planning. *Int. J. Robot. Res.* 30:846–94
36. Liniger A, Domahidi A, Morari M. 2014. Optimization-based autonomous racing of 1:43 scale RC cars. *Opt. Control Appl. Methods* 36:628–47
37. Andersen H, Schwarting W, Naser F, Eng YH, Ang MH Jr., et al. 2017. Trajectory optimization for autonomous overtaking with visibility maximization. In *2017 IEEE International Conference on Intelligent Transportation Systems (ITSC)*. New York: IEEE. In press
38. Kuwata Y, Teo J, Fiore G, Karaman S, Frazzoli E, How JP. 2009. Real-time motion planning with applications to autonomous urban driving. *IEEE Trans. Control Syst. Technol.* 17:1105–18
39. Tumova J, Hall GC, Karaman S, Frazzoli E, Rus D. 2013. Least-violating control strategy synthesis with safety rules. In *HSCC '13: Proceedings of the 16th International Conference on Hybrid Systems: Computation and Control*, pp. 1–10. New York: ACM
40. Vasile CI, Tumova J, Karaman S, Belta C, Rus D. 2017. Minimum-violation scLTL motion planning for mobility-on-demand. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1481–88. New York: IEEE
41. Janai J, Güney F, Behl A, Geiger A. 2017. Computer vision for autonomous vehicles: problems, datasets and state-of-the-art. arXiv:1704.05519
42. Geiger A, Lenz P, Stiller C, Urtasun R. 2013. Vision meets robotics: the KITTI dataset. *Int. J. Robot. Res.* 32:1231–37
43. Cordts M, Omran M, Ramos S, Rehfeld T, Enzweiler M, et al. 2016. The Cityscapes dataset for semantic urban scene understanding. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3213–23. New York: IEEE

44. Lowe DG. 1999. Object recognition from local scale-invariant features. In *Seventh IEEE International Conference on Computer Vision (ICCV)*, Vol. 2, pp. 1150–57. New York: IEEE
45. Lowe DG. 2004. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vis.* 60:91–110
46. Leutenegger S, Chli M, Siegwart RY. 2011. BRISK: Binary Robust Invariant Scalable Keypoints. In *2011 IEEE International Conference on Computer Vision (ICCV)*, pp. 2548–55. New York: IEEE
47. Bay H, Ess A, Tuytelaars T, Gool LV. 2008. Speeded-Up Robust Features (SURF). *Comput. Vis. Image Understand.* 110:346–59
48. Bay H, Tuytelaars T, Van Gool L. 2006. SURF: Speeded Up Robust Features. In *Computer Vision – ECCV 2006*, ed. A Leonardis, H Bischof, A Pinz, pp. 404–17. Berlin: Springer
49. Rublee E, Rabaud V, Konolige K, Bradski G. 2011. ORB: an efficient alternative to SIFT or SURF. In *2011 IEEE International Conference on Computer Vision (ICCV)*, pp. 2564–71. New York: IEEE
50. Mur-Artal R, Tardós JD. 2017. ORB-SLAM2: an open-source SLAM system for monocular, stereo, and RGB-D cameras. *IEEE Trans. Robot.* 33:1255–62
51. Zhang J, Singh S. 2015. Visual-LIDAR odometry and mapping: low-drift, robust, and fast. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2174–81. New York: IEEE
52. Forster C, Zhang Z, Gassner M, Werlberger M, Scaramuzza D. 2017. SVO: semidirect visual odometry for monocular and multicamera systems. *IEEE Trans. Robot.* 33:249–65
53. Engel J, Stckler J, Cremers D. 2015. Large-scale direct SLAM with stereo cameras. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1935–42. New York: IEEE
54. Bar Hillel A, Lerner R, Levi D, Raz G. 2014. Recent progress in road and lane detection: a survey. *Mach. Vis. Appl.* 25:727–45
55. Russakovsky O, Deng J, Su H, Krause J, Satheesh S, et al. 2015. ImageNet large scale visual recognition challenge. *Int. J. Comput. Vis.* 115:211–52
56. Ren S, He K, Girshick R, Sun J. 2017. Faster R-CNN: towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.* 39:1137–49
57. He K, Zhang X, Ren S, Sun J. 2016. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–78. New York: IEEE
58. Zhao H, Shi J, Qi X, Wang X, Jia J. 2017. Pyramid scene parsing network. arXiv:1612.01105
59. Paszke A, Chaurasia A, Kim S, Culurciello E. 2016. ENet: a deep neural network architecture for real-time semantic segmentation. arXiv:1606.02147
60. Zhao H, Qi X, Shen X, Shi J, Jia J. 2017. ICNet for real-time semantic segmentation on high-resolution images. arXiv:1704.08545
61. Ros G, Sellart L, Materzynska J, Vazquez D, Lopez AM. 2016. The SYNTHIA dataset: a large collection of synthetic images for semantic segmentation of urban scenes. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3234–43. New York: IEEE
62. Johnson-Roberson M, Barto C, Mehta R, Sridhar SN, Rosaen K, Vasudevan R. 2017. Driving in the matrix: Can virtual worlds replace human-generated annotations for real world tasks? In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 746–53. New York: IEEE
63. Richter SR, Vineet V, Roth S, Koltun V. 2016. Playing for data: ground truth from computer games. In *Computer Vision – ECCV 2016*, ed. B Leibe, J Matas, N Sebe, M Welling, pp. 102–18. Cham, Switz.: Springer
64. Herranz L, Jiang S, Li X. 2016. Scene recognition with CNNs: objects, scales and dataset bias. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 571–79. New York: IEEE
65. Gal Y, Ghahramani Z. 2016. Dropout as a Bayesian approximation: representing model uncertainty in deep learning. In *ICML '16: 33rd International Conference on Machine Learning*, ed. MF Balcan, KQ Weinberger, pp. 1050–59. New York: PMLR
66. McAllister R, Gal Y, Kendall A, van der Wilk M, Shah A, et al. 2017. Concrete problems for autonomous vehicle safety: advantages of Bayesian deep learning. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence (IJCAI-17)*, pp. 4745–53. Calif.: IJCAI
67. Chen C, Seff A, Kornhauser A, Xiao J. 2015. DeepDriving: learning affordance for direct perception in autonomous driving. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 2722–30. New York: IEEE

68. Caltagirone L, Bellone M, Svensson L, Wahde M. 2017. LIDAR-based driving path generation using fully convolutional neural networks. arXiv:1703.08987
69. Barnes D, Maddern W, Posner I. 2017. Find your own way: weakly-supervised segmentation of path proposals for urban autonomy. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 203–10. New York: IEEE
70. Pomerleau DA. 1989. ALVINN: an autonomous land vehicle in a neural network. In *Advances in Neural Information Processing Systems 1*, ed. DS Touretzky, pp. 305–13. San Francisco: Morgan Kaufmann
71. Muller U, Ben J, Cosatto E, Flepp B, Cun YL. 2006. Off-road obstacle avoidance through end-to-end learning. In *Advances in Neural Information Processing Systems 18*, ed. Y Weiss, PB Schölkopf, JC Platt, pp. 739–46. Cambridge, MA: MIT Press
72. Bojarski M, Del Testa D, Dworakowski D, Firner B, Flepp B, et al. 2016. End to end learning for self-driving cars. arXiv:1604.07316
73. Bojarski M, Yeres P, Choromanska A, Choromanski K, Firner B, et al. 2017. Explaining how a deep neural network trained with end-to-end learning steers a car. arXiv:1704.07911
74. Gurghian A, Koduri T, Bailur SV, Carey KJ, Murali VN. 2016. DeepLanes: end-to-end lane position estimation using deep neural networks. In *2016 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 38–45. New York: IEEE
75. Xu H, Gao Y, Yu F, Darrell T. 2016. End-to-end learning of driving models from large-scale video datasets. arXiv:1612.01079
76. Zhang J, Cho K. 2016. Query-efficient imitation learning for end-to-end autonomous driving. arXiv:1605.06450
77. Ross S, Gordon GJ, Bagnell D. 2011. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the 14th International Conference on Artificial Intelligence and Statistics*, pp. 627–35. New York: PMLR
78. Pfeiffer M, Schaeuble M, Nieto J, Siegwart R, Cadena C. 2017. From perception to decision: a data-driven approach to end-to-end motion planning for autonomous ground robots. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1527–33. New York: IEEE
79. Chen YF, Everett M, Liu M, How JP. 2017. Socially aware motion planning with deep reinforcement learning. arXiv:1703.08862
80. Richter C, Roy N. 2017. Safe visual navigation via deep learning and novelty detection. In *Robotics: Science and Systems XIII*, ed. N Amato, S Srinivasa, N Ayanian, S Kuindersma, chap. 64. N.p.: Robot. Sci. Syst. Found.
81. Wolf P, Hubschneider C, Weber M, Bauer A, Hrtl J, et al. 2017. Learning how to drive in a real world simulation with deep Q-networks. In *2017 IEEE Intelligent Vehicles Symposium (IV)*, pp. 244–50. New York: IEEE
82. You Y, Pan X, Wang Z, Lu C. 2017. Virtual to real reinforcement learning for autonomous driving. arXiv:1704.03952
83. Lillicrap TP, Hunt JJ, Pritzel A, Heess N, Erez T, et al. 2015. Continuous control with deep reinforcement learning. arXiv:1509.02971
84. Montemerlo M, Becker J, Bhat S, Dahlkamp H, Dolgov D, et al. 2008. Junior: the Stanford entry in the Urban Challenge. *J. Field Robot.* 25:569–97
85. Trautman P, Ma J, Murray RM, Krause A. 2015. Robot navigation in dense human crowds: statistical models and experimental studies of human-robot cooperation. *Int. J. Robot. Res.* 34:335–56
86. Toit NED, Burdick JW. 2012. Robot motion planning in dynamic, uncertain environments. *IEEE Trans. Robot.* 28:101–15
87. Sadigh D, Sastry S, Seshia SA, Dragan AD. 2016. Planning for autonomous cars that leverage effects on human actions. In *Robotics: Science and Systems XII*, ed. D Hsu, N Amato, S Berman, S Jacobs, chap. 29. N.p.: Robot. Sci. Syst. Found.
88. Kretzschmar H, Spies M, Sprunk C, Burgard W. 2016. Socially compliant mobile robot navigation via inverse reinforcement learning. *Int. J. Robot. Res.* 35:1289–307
89. Düring M, Pascheka P. 2014. Cooperative decentralized decision making for conflict resolution among autonomous agents. In *2014 IEEE International Symposium on Innovations in Intelligent Systems and Applications (INISTA) Proceedings*, pp. 154–61. New York: IEEE

90. Ulbrich S, Grossjohann S, Appelt C, Homeier K, Rieken J, Maurer M. 2015. Structuring cooperative behavior planning implementations for automated driving. In *2015 IEEE 18th International Conference on Intelligent Transportation Systems (ITSC)*, pp. 2159–65. New York: IEEE
91. Bahram M, Lawitzky A, Friedrichs J, Aeberhard M, Wollherr D. 2016. A game-theoretic approach to replanning-aware interactive scene prediction and planning. *IEEE Trans. Veh. Technol.* 65:3981–92
92. Lenz D, Kessler T, Knoll A. 2016. Tactical cooperative planning for autonomous highway driving using Monte-Carlo tree search. In *2016 IEEE Intelligent Vehicles Symposium (IV)*, pp. 447–53. New York: IEEE
93. Schwarting W, Pascheka P. 2014. Recursive conflict resolution for cooperative motion planning in dynamic highway traffic. In *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 1039–44. New York: IEEE
94. Li N, Oyler DW, Zhang M, Yildiz Y, Kolmanovsky I, Girard AR. 2017. Game theoretic modeling of driver and vehicle interactions for verification and validation of autonomous vehicle control systems. *IEEE Trans. Control Syst. Technol.* In press. **https://doi.org/10.1109/TCST.2017.2723574**
95. Wei J, Dolan JM, Litkouhi B. 2013. Autonomous vehicle social behavior for highway entrance ramp management. In *2013 IEEE Intelligent Vehicles Symposium (IV)*, pp. 201–7. New York: IEEE
96. Evestedt N, Ward E, Folkesson J, Axehill D. 2016. Interaction aware trajectory planning for merge scenarios in congested traffic situations. In *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, pp. 465–72. New York: IEEE
97. Hoermann S, Stumper D, Dietmayer K. 2017. Probabilistic long-term prediction for autonomous vehicles. In *2017 IEEE Intelligent Vehicles Symposium (IV)*, pp. 237–43. New York: IEEE
98. Dong C, Dolan JM, Litkouhi B. 2017. Intention estimation for ramp merging control in autonomous driving. In *2017 IEEE Intelligent Vehicles Symposium (IV)*, pp. 1584–89. New York: IEEE
99. Hubmann C, Becker M, Althoff D, Lenz D, Stiller C. 2017. Decision making for autonomous driving considering interaction and uncertain prediction of surrounding vehicles. In *2017 IEEE Intelligent Vehicles Symposium (IV)*, pp. 1671–78. New York: IEEE
100. Liu W, Kim SW, Pendleton S, Ang MH. 2015. Situation-aware decision making for autonomous driving on urban road using online POMDP. In *2015 IEEE Intelligent Vehicles Symposium (IV)*, pp. 1126–33. New York: IEEE
101. Ulbrich S, Maurer M. 2015. Towards tactical lane change behavior planning for automated vehicles. In *2015 IEEE 18th International Conference on Intelligent Transportation Systems (ITSC)*, pp. 989–95. New York: IEEE
102. Galceran E, Cunningham AG, Eustice RM, Olson E. 2017. Multipolicy decision-making for autonomous driving via changepoint-based behavior prediction: theory and experiment. *Auton. Robots* 41:1367–82
102a. Zhou B, Schwarting W, Rus D, Alonso-Mora J. 2018. Joint multi-policy behavior estimation and receding-horizon trajectory planning for automated urban driving. In *2018 IEEE International Conference on Robotics and Automation (ICRA).* New York: IEEE. In press
103. Brechtel S, Gindele T, Dillmann R. 2014. Probabilistic decision-making under uncertainty for autonomous driving using continuous POMDPs. In *2014 IEEE 17th International Conference on Intelligent Transportation Systems (ITSC)*, pp. 392–99. New York: IEEE
104. Shalev-Shwartz S, Shammah S, Shashua A. 2016. Safe, multi-agent, reinforcement learning for autonomous driving. arXiv:1610.03295
105. Vallon C, Ercan Z, Carvalho A, Borrelli F. 2017. A machine learning approach for personalized autonomous lane change initiation and control. In *2017 IEEE Intelligent Vehicles Symposium (IV)*, pp. 1590–95. New York: IEEE
106. Lenz D, Diehl F, Le MT, Knoll A. 2017. Deep neural networks for Markovian interactive scene prediction in highway scenarios. In *2017 IEEE Intelligent Vehicles Symposium (IV)*, pp. 685–92. New York: IEEE
107. Lee N, Choi W, Vernaza P, Choy CB, Torr PH, Chandraker M. 2017. DESIRE: distant future prediction in dynamic scenes with interacting agents. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2165–74. New York: IEEE

108. Sadigh D, Sastry SS, Seshia SA, Dragan A. 2016. Information gathering actions over human internal state. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 66–73. New York: IEEE
109. Sadigh D, Dragan A, Sastry S, Seshia S. 2017. Active preference-based learning of reward functions. In *Robotics: Science and Systems XIII*, ed. N Amato, S Srinivasa, N Ayanian, S Kuindersma, chap. 53. N.p.: Robot. Sci. Syst. Found.
110. Huang SH, Held D, Abbeel P, Dragan AD. 2017. Enabling robots to communicate their objectives. In *Robotics: Science and Systems XIII*, ed. N Amato, S Srinivasa, N Ayanian, S Kuindersma, chap. 59. N.p.: Robot. Sci. Syst. Found.
111. Abbeel P, Ng AY. 2004. Apprenticeship learning via inverse reinforcement learning. In *ICML '04: Proceedings of the Twenty-First International Conference on Machine Learning*, chap. 1. New York: ACM
112. Abbeel P, Dolgov D, Ng AY, Thrun S. 2008. Apprenticeship learning for motion planning with application to parking lot navigation. In *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1083–90. New York: IEEE
113. Ziebart BD, Maas AL, Bagnell JA, Dey AK. 2008. Maximum entropy inverse reinforcement learning. In *23rd AAAI Conference on Artificial Intelligence*, pp. 1433–38. Menlo Park, CA: AAAI Press
114. Kuderer M, Gulati S, Burgard W. 2015. Learning driving styles for autonomous vehicles from demonstration. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2641–46. New York: IEEE
115. Pfeiffer M, Schwesinger U, Sommer H, Galceran E, Siegwart R. 2016. Predicting actions to act predictably: cooperative partial motion planning with maximum entropy models. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 2096–101. New York: IEEE
116. Herman M, Fischer V, Gindele T, Burgard W. 2015. Inverse reinforcement learning of behavioral models for online-adapting navigation strategies. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3215–22. New York: IEEE
117. Ratliff ND, Bagnell JA, Zinkevich MA. 2006. Maximum margin planning. In *Proceedings of the 23rd International Conference on Machine Learning*, pp. 729–36. New York: ACM
118. Silver D, Bagnell JA, Stentz A. 2010. Learning from demonstration for autonomous navigation in complex unstructured terrain. *Int. J. Robot. Res.* 29:1565–92
119. Silver D, Bagnell JA, Stentz A. 2013. Learning autonomous driving styles and maneuvers from expert demonstration. In *Experimental Robotics: The 13th International Symposium on Experimental Robotics*, ed. J Desai, G Dudek, O Khatib, V Kumar, pp. 371–86. Heidelberg, Ger.: Springer
120. Levine S, Koltun V. 2012. Continuous inverse optimal control with locally optimal examples. In *Proceedings of the 29th International Conference on International Conference on Machine Learning*, pp. 475–82. Madison, WI: Omnipress
121. Majumdar A, Singh S, Mandlekar A, Pavone M. 2017. Risk-sensitive inverse reinforcement learning via coherent risk models. In *Robotics: Science and Systems XIII*, ed. N Amato, S Srinivasa, N Ayanian, S Kuindersma, chap. 69. N.p.: Robot. Sci. Syst. Found.
122. Wulfmeier M, Ondruska P, Posner I. 2015. Maximum entropy deep inverse reinforcement learning. arXiv:1507.04888
123. Wulfmeier M, Rao D, Wang DZ, Ondruska P, Posner I. 2017. Large-scale cost function learning for path planning using deep inverse reinforcement learning. *Int. J. Robot. Res.* 10:1073–87
124. Kuefler A, Morton J, Wheeler T, Kochenderfer M. 2017. Imitating driver behavior with generative adversarial networks. arXiv:1701.06699
125. Ho J, Ermon S. 2016. Generative adversarial imitation learning. In *Advances in Neural Information Processing Systems 29*, ed. DD Lee, M Sugiyama, UV Luxburg, I Guyon, R Garnett, pp. 4565–73. New York: Curran Assoc.
126. Kalra N, Paddock S. 2016. *Driving to safety: How many miles of driving would it take to demonstrate autonomous vehicle reliability?*. Rep. RR-1478-RC, Rand Corp., Santa Monica, CA. **http://www.rand.org/pubs/research_reports/RR1478.html**
127. Nilsson P, Hussien O, Balkan A, Chen Y, Ames AD, et al. 2016. Correct-by-construction adaptive cruise control: two approaches. *IEEE Trans. Control Syst. Technol.* 24:1294–307

128. Kim ES, Arcak M, Seshia SA. 2015. Compositional controller synthesis for vehicular traffic networks. In *2015 54th IEEE Conference on Decision and Control (CDC)*, pp. 6165–71. New York: IEEE
129. Wongpiromsarn T. 2010. *Formal methods for design and verification of embedded control systems: application to an autonomous vehicle*. PhD Thesis, Calif. Inst. Technol., Pasadena, CA
130. Loos SM, Platzer A, Nistor L. 2011. Adaptive cruise control: hybrid, distributed, and now formally verified. In *FM 2011: Formal Methods*, ed. M Butler, W Schulte, pp. 42–56. Berlin: Springer
131. Althoff M, Dolan JM. 2014. Online verification of automated road vehicles using reachability analysis. *IEEE Trans. Robot.* 30:903–18
132. Schürmann B, Heß D, Eilbrecht J, Stursberg O, Köster F, Althoff M. 2017. Ensuring drivability of planned motions using formal methods. In *2017 20th IEEE Intelligent Transportation Systems Conference (ITSC)*. New York: IEEE. In press
133. Liebenwein L, Schwarting W, Vasile CI, DeCastro J, Alonso-Mora J, et al. 2018. Compositional and contract-based verification for autonomous driving on road networks. In *Robotics Research: The 18th International Symposium ISRR*. Forthcoming
134. Katz G, Barrett C, Dill DL, Julian K, Kochenderfer MJ. 2017. Reluplex: an efficient SMT solver for verifying deep neural networks. In *Computer Aided Verification: 29th International Conference, CAV 2017, Heidelberg, Germany, July 24–28, 2017, Proceedings, Part I*, ed. R Majumdar, V Kunčak, pp. 97–117. Cham, Switz.: Springer
135. Seshia SA, Sadigh D, Sastry SS. 2016. Towards verified artificial intelligence. arXiv:1606.08514
136. Pavone M, Smith S, Frazzoli E, Rus D. 2012. Robotic load balancing for mobility-on-demand systems. *Int. J. Robot. Res.* 31:839–54
137. Zhang R, Pavone M. 2016. Control of robotic mobility-on-demand systems: a queueing-theoretical perspective. *Int. J. Robot. Res.* 35:186–203
138. de Almeida Correia GH, van Arem B. 2016. Solving the user optimum privately owned automated vehicles assignment problem (UO-POAVAP): a model to explore the impacts of self-driving vehicles on urban mobility. *Transp. Res. B* 87:64–88
139. Toth P, Vigo D. 2014. *Vehicle Routing: Problems, Methods, and Applications*. Philadelphia: SIAM. 2nd ed.
140. Pillac V, Gendreau M, Guéret C, Medaglia AL. 2013. A review of dynamic vehicle routing problems. *Eur. J. Oper. Res.* 225:1–11
141. Agatz NA, Erera AL, Savelsbergh MW, Wang X. 2011. Dynamic ride-sharing: a simulation study in metro Atlanta. *Transp. Res. B* 45:1450–64
142. Santi P, Resta G, Szell M, Sobolevsky S, Strogatz SH, Ratti C. 2014. Quantifying the benefits of vehicle pooling with shareability networks. *PNAS* 111:13290–94
143. Alonso-Mora J, Samaranayake S, Wallar A, Frazzoli E, Rus D. 2017. On-demand high-capacity ride-sharing via dynamic trip-vehicle assignment. *PNAS* 114:462–67
144. Shaheen S, Christensen M. 2014. The true future of transportation has two big barriers to entry. *CityLab*, Apr. 25. **https://www.citylab.com/transportation/2014/04/true-future-transportation-has-two-big-barriers-entry/8933**
145. NYC OpenData. 2016. *New York City yellow taxi trip data*. **https://data.cityofnewyork.us/dataset/2016-Yellow-Taxi-Trip-Data/k67s-dv2t**
146. Ritzinger U, Puchinger J, Hartl RF. 2016. A survey on dynamic and stochastic vehicle routing problems. *Int. J. Prod. Res.* 54:215–31
147. Alonso-Mora J, Wallar A, Rus D. 2017. Predictive routing for autonomous mobility-on-demand systems with ride-sharing. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 3583–90. New York: IEEE
148. Barnard M. 2016. Autonomous cars likely to increase congestion. *CleanTechnica*, Jan. 17. **http://cleantechnica.com/2016/01/17/autonomous-cars-likely-increase-congestion**
149. Zhang R, Rossi F, Pavone M. 2017. Routing autonomous vehicles in congested transportation networks: structural properties and coordination algorithms. *Auton. Robots.* In press
150. Levin MW. 2017. Congestion-aware system optimal route choice for shared autonomous vehicles. *Transp. Res. C* 82:229–47
151. de Weerdt MM, Stein S, Gerding EH, Robu V, Jennings NR. 2016. Intention-aware routing of electric vehicles. *IEEE Trans. Intell. Transp. Syst.* 17:1472–82

Annual Review of Control, Robotics, and Autonomous Systems

Volume 1, 2018

Contents

Errata

An online log of corrections to *Annual Review of Control, Robotics, and Autonomous Systems* articles may be found at http://www.annualreviews.org/errata/control