2019 IEEE Intelligent Transportation Systems Conference (ITSC)
Auckland, NZ, October 27-30, 2019

Evolutionary Learning in Decision Making for Tactical Lane Changing

Tingting Li, Jianping Wu, Ching-Yao Chan

Abstract—**Decision making for lane change is an important challenge for automated vehicles, especially in complex traffic environments. In recent years, there have been studies that utilize reinforcement learning for lane change applications. However, such an approach requires high computational costs and is difficult to implement by parallel computing. To overcome the problem, an evolutionary learning approach is put forward for the decision-making application of autonomous driving. By deploying the parallel workers, making parameters of the neural network mutate and recombining the well-behaved off-springs during the evolutionary learning process, the Evolution Strategy (ES) agent learns to make decisions for lane-change maneuvers. At the same time, safety verification is performed, which ensures driving safety and simplifies the learning process. To test the performance of the proposed method, a highway simulation environment is established. The results show that the combination of the high-level evolutionary learning and low-level safety verification jointly achieve efficient driving behavior control.**

***Key words*—Evolutionary learning, Lane change, Decision making**

I. Introduction

One key challenge in developing autonomous driving is the behavioral decision making in complex traffic environments. Inappropriate lane changes are demonstrated to cause about 4% to 10% traffic accidents [1]. Besides, these lane-change maneuvers and incidents are regarded as the anomaly that potentially impacts safety and interferes with traffic flow by causing the propagation of shock wave [2]. To realize efficient decision making for driving, the real-time surrounding traffic environment should be monitored and modelled, and the probable trajectories of various traffic participants should be predicted. Thus, developing an efficient decision-making process for lane change maneuvering is highly desirable, by exploring the environment and learning from the experience automatically.

Rule-based models are commonly used for the task of decision making for lane change maneuvers, in which the automated vehicle changes lane according to specific rules such as the gap acceptance model [3], the minimize-overall-braking induced by lane changes (MOBIL) model [4] and Cellular Automata Model [5]. However, vehicles using rule-based models are too cautious in the various and complex traffic situations, which leads to the unnecessary delay. Another approach is the application of the game theory model, in which the problem is solved where the automated vehicle interacts with other vehicles in a way that maximizes the expected reward [6]. Nevertheless, there is a limited number of vehicles which are considered in the game theory. The utility-based model is also applied to maximize the lane change utility based on the partially observable Markov decision process (POMDP) [7], which requires the complex calculation. Thus, the common problems of aforementioned models are that they are aimed at a certain traffic scene and a complex procedure is required for interpreting a different scene. Compared with these models, the deep learning methods have advantages of exploring the environment and learning from the experience, which is suitable for overcoming these problems.

Deep learning algorithms have been used in the autonomous driving successfully. For instance, the bidirectional Recurrent Neural Network (RNN) is conducted to assess the traffic scene and classify the driving situation [8]. The deep convolutional neural network (DCNN) is trained to classify whether the adjacent lane is blocked or free based on the rear side view images [9]. The Inverse Reinforcement Learning (IRL) model is implemented to extract the individual driving style and plan the driving trajectory [10]. The Deep Deterministic Policy Gradient (DDPG) model is used to learn the driving maneuver for the overtaking and car-following operation [11, 12]. These works lay a foundation for the behavior control of the automated vehicle by deep learning.

Based on a literature review of the lane-change decision-making problem, it can be found that the Reinforcement Learning (RL) method has been applied in this field, particularly in the last few years. The Deep Q-Network agent and Deep Deterministic Actor Critic (DDAC) agent have been introduced to handle the speed control problem and the on-ramp merging problem for automated vehicles [13]. In the specific lane-change scenario, the deep Q-learning algorithm is also adopted to learn to make decisions about acceleration and deceleration [14]. Besides, the rule-based control is combined with the deep Q-learning algorithm to achieve a faster learning rate [15]. The existing research has demonstrated the ability of these learning approaches for achieving safe lane change. However, owing to the feature of Q-learning and policy gradient optimization method, the full gradient must be communicated across different processes, which causes a high communicational cost during the training stage. Besides, the gradient estimate may be biased when the action has long-lasting effect, which obstructs the learning process.

To overcome these limitations of the existing work, we propose a decision-making method for lane change based on

Tinging Li is with the Civil Engineering Department, Tsinghua University, Beijing, 100084 CHINA (e-mail: ttlithu@163.com).

Jianping Wu is with the Civil Engineering Department, Tsinghua University, Beijing, 100084 CHINA (e-mail: jianpingwu@tsinghua.edu.cn).

Ching-Yao Chan is with California PATH, University of California, Berkeley, CA 94804 USA (e-mail: cychan@berkeley.edu).

978-1-5386-7024-8/19/$31.00 ©2019 IEEE

the Evolutionary Strategy (ES) [16]. To our best knowledge, this is the first application of the ES algorithm in the field of decision-making research for lane change. ES is a kind of black box optimization algorithm, which imitates the natural evolution procedure: the different generation is regarded as the iteration, the parameter vectors is regarded as the genomes, and the perturbation of the parameter is treated as the mutation [16]. After the generation is mutated, the well-behaved offspring is selected and recombined to produce the next offspring. The evolution procedure will be repeated until the objective is totally optimized. Compared with the Q-learning and policy gradient optimization models, the evolution term is independent of the time step, thus the ES algorithm is effective when the action has long-lasting effects. Additionally, the communication is only focused on the scalar return and random seed rather than the full gradient. As a result, the ES algorithm is highly parallelizable and has fewer hyperparameters. Conceptually, the ES algorithm seems like the action of hill-climbing in a high-dimensional space relying on the finite variances along several random directions at each time step.

The ES algorithm belongs to the evolutionary algorithms (EA). The tests of the Atari and MoJoCo tasks show that the ES model can make up for the decreased data efficiency by deploying the parallel workers [17]. The experiment of the networked evolutionary algorithm also displays that the sparser communication between learning agents causes the higher learning rate [18].

In this paper, we put forward a two-stage lane-change decision making model. In the first stage, the ES-based approach is applied to learn the high-level lane change decision making for the automated vehicle. In the second stage, the low-level safety verification is performed to ensure safe driving. The proposed model is able to

- learn the lane-change decision making in different surrounding environment by deploying the parallel workers and executing the evolutionary learning process;
- combine the high-level learning and low-level safety verification to achieve efficient driving behavior control;
- be extended to different traffic scenes by adjusting the state space, action space, reward function and the safety verification rule.

The reminder of this paper is organized as follows: Section II introduces the definition of the problem. Section III presents the ES algorithm. Section IV provides the description of the safety verification rule. Section V depicts the algorithm application and simulation experiment. Section VI states the result and analysis. Section VII describes the conclusions.

II. Problem

The lane-change problem is shown in a configuration illustrated in Figure 1. The ego vehicle in the middle lane senses the information of the surrounding environment and learns to drive efficiently. For the purpose of discussions in this paper, we assume that the lane change action is controlled by a two-stage setup, a high-level decision making and a low-level safety verification. At the higher level, the ego vehicle is trained to learn to make decisions so as to drive as fast as possible and minimize the disruption to the traffic flow. At the same time, the safety verification stage guarantees that the agent only chooses the safe action.

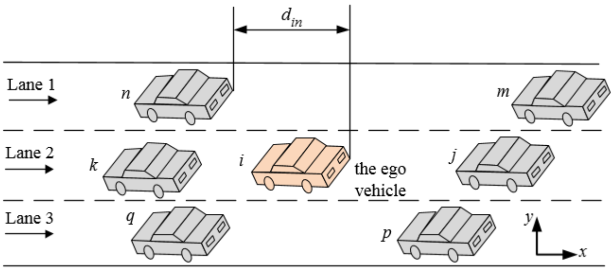


Fig. 1. The surrounding environment of the ego vehicle *i*.

A. State Representation

In Figure 1, the coordinate of the vehicle *i* is (x_i, y_i). And the longitudinal velocity of vehicle *i* is v_i. The vehicle *j* and vehicle *k* are the leading vehicle and following vehicle on the current lane. Vehicle *m* and *n* are the leading vehicle and following vehicle on the left lane. Vehicles *p* and *q* are the leading vehicle and following vehicle on the right lane. The state representation contains the relative velocity and distance information, which consists of 13 variables:

$$\mathrm{s} = (d_{ij}, d_{im}, d_{ip}, v_i, v_{ij}, v_{im}, v_{ip}, d_{ik}, d_{in}, d_{iq}, v_{ik}, v_{in}, v_{iq}) \qquad (1)$$

Variables d_{ij} and v_{ij} are the relative distance and velocity between the ego vehicle i and the vehicle j. The values of d_{ij} and v_{ij} are calculated as $d_{ij} = y_i - y_j$ and $v_{ij} = v_i - v_j$. The values of different variables are scaled in the range between -1 and 0 before training. When there is no leading vehicle, the corresponding relative distance and the relative velocity are both assigned as -1. When there is no following vehicle, the corresponding relative distance and the relative velocity are both assigned as 1. The action of the ego vehicle is influenced by leading vehicles and following vehicles on the current lane and adjacent lanes. Thus, the state space is related to the information of the ego vehicle and surrounding vehicles.

B. Action Representation

The decision-making model focuses on whether to change lanes and the selection of the target lane. The ego vehicle has three available control actions at each time step:

- *a$_{1,}$* stay on the current lane
- *a$_{2,}$* make a lane change to the left
- *a$_{3,}$* make a lane change to the right

The task is to learn to select one of the actions in the discrete action space. To make sure that only safe action is executed, the safety verification is performed before the discrete action is conducted.

C. *Reward Function*

The learning goal of the ego vehicle is to drive as fast as possible and to minimize the disruption to the following vehicle at the same time. On the one hand, the surrounding traffic environment has an influence on the velocity. If the gap between the ego vehicle and the leading vehicle on the current lane is large enough, then the ego vehicle can accelerate to the desired speed and do not need to change lane. Thus, the reward function is affected by the difference between the real-time velocity and the desired velocity. On the other hand, the frequent tactical lane change might induce stop-and-go waves. Hence, the reward function is related to the acceleration of the following vehicle on the current lane. As (2) shows, the reward r_t reflects the goal of driving. Variable $v_{i,t}$ is the velocity of the ego vehicle at time instant *t*. v_{desire} is the desired velocity of the ego vehicle. $a_{k,t}$ is the acceleration of the following vehicle on the current lane.

$$r_t = -\left|v_{i,t} - v_{desire}\right| + a_{k,t} \tag{2}$$

III. Evolution Strategy

There are several evolution algorithms, including the Covariance Matrix Adaptation Evolution Strategy (CMAES), Neuro Evolution of Augmenting Topologies (NEAT) and Natural Evolution Strategy (NES). These algorithms differ in the mechanism to generate the individuals in a generation. In this paper, we use the version of NES, which has been implemented into the RL benchmark problems by OpenAI [17]. As Figure 2 shows, the principle of the ES algorithm is injecting the noise in the parameter space rather than the action space like a RL-based model. As a result, ES is the "guess and check" on parameters. It is also possible to add noise in both actions and parameters to potentially combine the RL-based methods and ES-based methods.

In the ES algorithm, the parameter θ means the weight of the network. It is drawn from the distribution $p_\varphi(\theta)$, where the parameter φ is searched to maximize the average expected fitness $E_{\theta\sim p_\varphi}F(\theta)$. The function $F(\theta)$ is applied to evaluate the variable θ. In this paper, the population distribution is instantiated by the isotropic multivariate Gaussian distribution $N(\varphi,\delta^2 I)$. Then the expected fitness is depicted as $E_{\varepsilon\sim N(0,I)}F(\theta+\delta\varepsilon)$. The function is optimized by the stochastic gradient ascent method:

$$\nabla E_{\varepsilon\sim N(0,I)}F(\theta+\delta\varepsilon) = \frac{1}{\delta}E_{\varepsilon\sim N(0,I)}\{F(\theta+\delta\varepsilon)\varepsilon\} \tag{3}$$

In the generation iteration, the sample of the population is drawn from $N(0,I)$. The parameter θ is updated according to (4). The variable n means the population size, and α represents the step size.

$$\theta \leftarrow \theta + \alpha\frac{1}{n\delta}\sum_{i=1}^{n}F(\theta+\delta\varepsilon_i)\varepsilon_i \tag{4}$$

The detailed algorithm is depicted in Algorithm 1. In the ES algorithm, workers take advantage of the shared random seeds, which reduces the communication cost. Besides, the mirrored sampling of the parameter ε is executed, which reduces the probability of falling into the local optima and the influence of the outlier individuals.

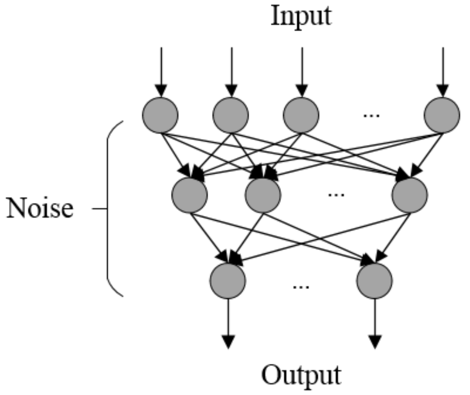


Fig. 2. The neural network example of ES.

Algorithm 1: Evolution Strategy

Input: Learning rate α, noise standard deviation δ, initial policy parameter θ, n individual workers with known random seeds

```
while the termination criterion is not fulfilled do
    $t \leftarrow t+1$
    for i=1,2, …, n do
        generate $\varepsilon_i$, which is drawn from $N(0,I)$
        evaluate the fitness function values $F_i = F(\theta_t + \delta\varepsilon_i)$
    end
    workers send the scalar values of $F_i$ to each other
    for i=1,2, …, n do
        for j=1,2, …, n do
            reconstruct the values of $\varepsilon_j$ with known
            random seeds
        end
        update $\theta_{t+1} \leftarrow \theta_t + \alpha\frac{1}{n\delta}\sum_{j=1}^{n}F_j\varepsilon_j$
    end
end
```

IV. Safety Verification

In our proposed approach, we combine the low-level safety verification with the high-level action learning. The safety verification ensures that only safe action is performed, then the prior knowledge is incorporated into the model and the reward function is simplified. Thus, the agent is able to concentrate on acquiring a faster speed. The verification rules contain the suitable lane rule and the safe gap rule.

A. *Suitable Lane*

This verification process guarantees that only the available lane is chosen, which is expressed in Table I. Variables a_1, a_2

TABLE I. FEASIBLE VALUE OF ACTION

Action Value	$l_i = 1$	$l_i = 2$	$l_i = 3$
a_1	√	√	√
a_2	×	√	√
a_3	√	√	×

and a_3 are available actions. l_i is the current lane number of the ego vehicle. The sign "true" means that the action can be executed when the ego vehicle is on the corresponding lane. The sign "false" is the limit of the action. For instance, if the ego vehicle is on the Lane 1, it will not be allowed to change lane to the left as there is no lane on the left side. This rule helps the agent to avoid driving off the road.

B. Safe Gap between Vehicles

The gap between the ego vehicle and its leading and following vehicles on the target lane should be large enough to keep safe. The safe gap rule is represented as (5) and (6). $x_{ego}(t)$ is the longitudinal coordinate of the ego vehicle at time t. Variables $x_{leading}(t)$ and $x_{following}(t)$ are the longitudinal coordinates of the leading and following vehicles on the target lane. The variable $x^{l}_{safe}(t)$ is the safe distance between the ego vehicle and the leading vehicle. The variable $x^{f}_{safe}(t)$ is the safe distance [19] between the ego vehicle and the following vehicle. The ego vehicle will only choose the action which meets the safe gap rule.

$$\left| x_{ego}(t) - x_{leading}(t) \right| \geq x^{l}_{safe}(t) \tag{5}$$

$$\left| x_{ego}(t) - x_{following}(t) \right| \geq x^{f}_{safe}(t) \tag{6}$$

Except the safe gap rule, the Gipps car-following model [20] is applied to avoid collisions. In the proposed model, if the action is verified to be unsafe with these rules, then it will not be executed. Thus, collisions are never allowed during the model training procedure, which simplifies the learning process. The modularized model has been demonstrated to behave better than the end-to-end model in the autonomous driving [21].

The proposed learning model can be apllied to different traffic scenes. If there is another supplementary training goal, the training can be accomplished by adjusting the state space, action space, reward function and the safety verification rule.

V. ALGORITHM APPLICATION AND EXPERIMENT

A. Algorithm Application

In this paper, the lane change action is executed by relying on the high-level decision-making and the low-level safety verification. As Figure 3 shows, there are totally four fully connected layers in the neural network architecture. And the numbers of neurons are 13, 350, 300 and 3 for the individual layer. The detailed parameter of the network is depicted in Table II. The thirteen features of the state are the inputs to the model, which capture the distance and velocity information of the ego vehicle, the leading vehicles and following vehicles. The discrete lane change choice is the output of the network.

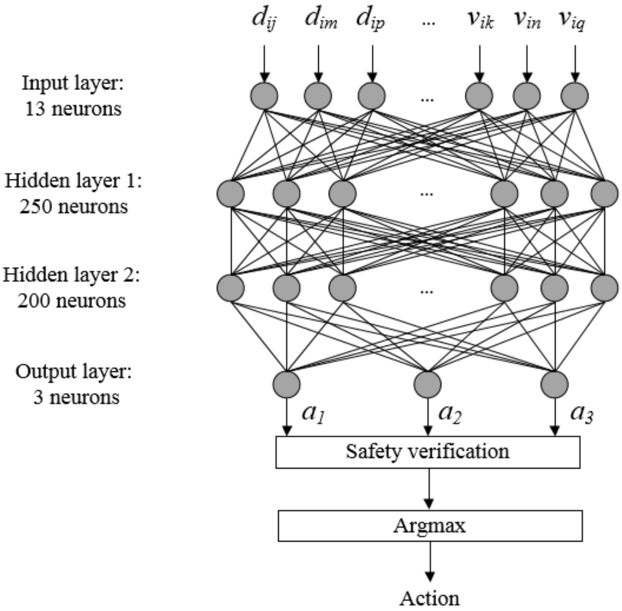


Fig. 3. The decision-making architecture of the proposed model.

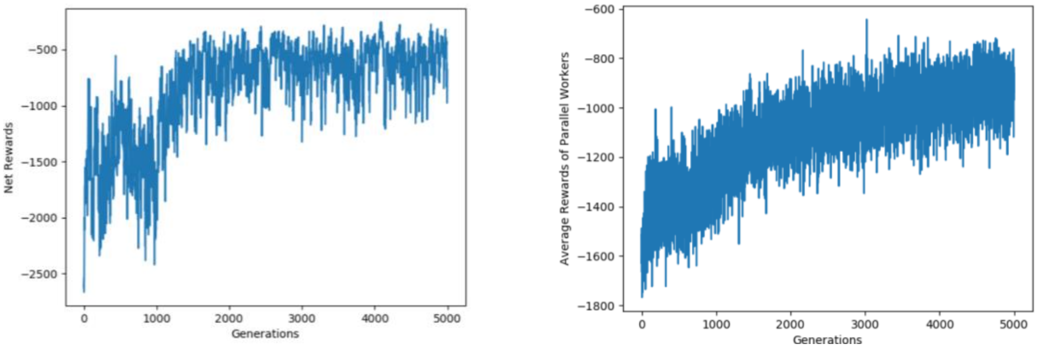
TABLE II. PARAMETER OF THE ES ALGORITHM

Parameter	*Description*
Number of input neurons	13
Number of output neurons	3
Number of hidden layers	2
Number of neurons in hidden layers	350,300
Learning rate	0.05
Mutation rate	0.05
Training population	160

The low-level safety verification is performed for each action. The final action is assigned as the action which meets the safety rule and has the maximum output value.

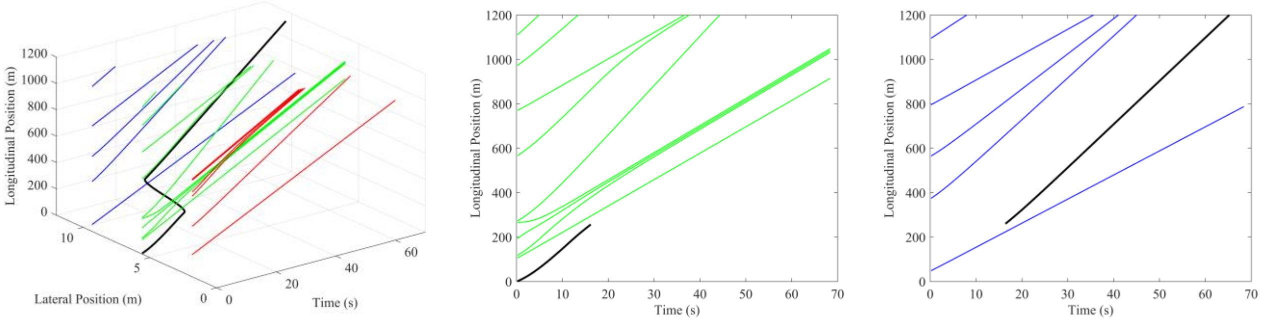
B. Simulation Experiment

In order to evaluate of the proposed algorithm, simulated experiments are performed. The road with three lanes on the highway is designed, and the width of each lane is 3.75m. The total length of the road is 1.2 km. Except for the ego vehicle, there are 20 other vehicles on the road. All vehicles are assumed to be the same length and width. The desired longitudinal velocity of the ego vehicle is 19.5m/s. The desired longitudinal velocities of other vehicles are assigned as the arbitrary values in the range of 10m/s-24m/s. The initial velocities of all vehicles are assigned as the value between 10m/s and the individual desired velocities. The Gipps car-following model is applied to keep the safe car-following behavior. The initial longitudinal and lateral coordinates of the ego vehicle are 0 and 5.625m, i.e., the initial location of the ego vehicle is on the starting point of the Lane 2. The longitudinal coordinates of other vehicles are assigned randomly between 0-1.2 km. And the lane number of other vehicles are also initialized randomly. The lane change time of



(a) (b)

Fig. 4. Training results of the model. (a) Rewards of the network. (b) Average rewards of parallel workers.



(a) (b) (c)

Fig. 5. Trajectories of the ego vehicle and other vehicles. (a) Trajectories of all vehicles in the three-dimensional space. (b) Longitudinal trajectories of vehicles on Lane 2. (c) Longitudinal trajectories of vehicles on Lane 1.

the ego vehicle is defined as 3.6s. When the ego vehicle arrives at the end of the road, the experiment is terminated and repeated. As for the ES algorithm, the learning rate is set as 0.05. As Table II shows, the population size is 160. And the mutation rate of each generation is 0.05.

VI. Results and Analysis

In model training, 5000 evolutionary generations are implemented. The rewards of the network during training are depicted in Figure 4 (a). It can be observed that the curve of the reward rises upward and converges to a value around -500, implying that the agent learns to take actions with higher rewards. As Figure 4 (b) shows, the average rewards of the parallel workers also increase with the evolution to later generations. The convergence trend is shown after 2000 generations. The training results demonstrate that the proposed lane-change model is feasible.

In order to test the effectiveness of the final lane-change model, the detailed information of the ego vehicle and other vehicles are analyzed in the random experiment. In Figure 5(a), trajectories of all vehicles are shown in the three-dimensional space. The *x*, *y* and *z* coordinates represent the lateral positions, time and the longitudinal positions. The blue lines, green lines and red lines are trajectories of vehicles on the Lane 1, Lane 2 and Lane 3, respectively. The black line shows the trajectory of the ego vehicle. It can be seen that the ego vehicle changes lane from the Lane 2 to the Lane 1 during the experiment. To analyze the information of the surrounding environment, the longitudinal trajectories of vehicles on Lane 2 and Lane 1 are depicted in Figure 5 (b) and Figure 5 (c). As Figure 5 (b) and Figure 5 (c) show, the black lines are trajectories of the ego vehicle on the Lane 2 and Lane1. The green ones and blue ones correspond to vehicle trajectories on Lane 2 and Lane 1 separately. It can be seen that in the first 16.2 seconds, the ego vehicle stays on the Lane 2. When it gets closer and closer to the leading vehicle, it tends to find a bigger leading gap and changes to Lane 1. As Figure 6 shows, the longitudinal velocity of the ego vehicle increases during the first 11.6 seconds. When the ego vehicle gets closer to the leading vehicle, it decelerates sharply. At around the 18th second, the ego vehicle starts to accelerate, which corresponds to the complete of the lane change. Based on Figure 5 (b) and Figure 5 (c), it can be concluded that when the environment presents hinderance on the driving, the ego vehicle will change lane and get a better driving environment for its goal,

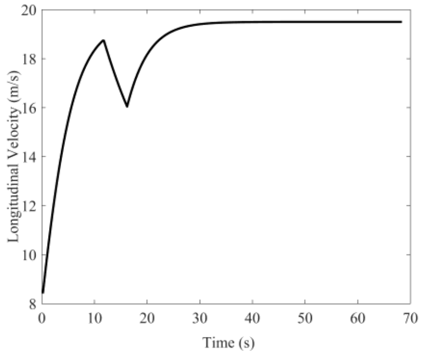


Fig. 6. The longitudinal velocity of the ego vehicle.

which is to achieve a higher speed and minimize the disruption to the following vehicle. The driving behavior of the ego vehicle reflects our deign of the reward. From this perspective, the proposed algorithm has achieved the goal of the proposed model and leads to the efficient driving.

VII. Conclusion and Discussion

In this paper, a decision-making model for lane change based on a high-level evolutionary learning and a low-level safety verification is proposed. To our best knowledge, this is the first application of the ES in the research domain of lane-change decision making. A deep neural network structure is designed, and parameters are mutated and evolved in the proposed model. The model training results indicate that the agent is able to achieve a fast learning rate. Besides, the progression of rewards shows a convergence trend after 1800 generations. The simulation experiment also shows that the lane-change behavior control ensures the ego vehicle achieve safe and efficient driving, which lays a foundation for the real-time application of the model.

The proposed method provides an innovative way of solving the decision-making problem for lane change maneuvers of the autonomous driving. One of the topics for future studies will be to conduct a comparative evaluation of the proposed model and the RL-based model. Besides, the combination of ES-based model and RL-based model can be explored, that is, the disturbances of the action space and the parameter space exist together. Furthermore, the model can be extended to test in more complex traffic environments. For instance, when an ego vehicle is approaching an exit of a freeway, the priority of the ego vehicle to exit and its maneuver to the ramp should be considered. Moreover, the interaction between human-driven vehicles and automated vehicles and the prediction of human driver behavior should also be taken into consideration.

Acknowledgment

This research is supported in part by the National Natural Science of Foundation of China (NSFC) under the Grant No. U1709212 and No. U1509205. This research is also partially supported by China Association for Science and Technology: Research on frontiers of intelligent transport system.

References

[1] T. v. Dijck and G. A. J. v. d. Heijden, "VisionSense: an advanced lateral collision warning system," in *IEEE Proceedings. Intelligent Vehicles Symposium, 2005.*, 2005, pp. 296-301.

[2] T. Awal, M. Murshed, and M. Ali, "An efficient cooperative lane-changing algorithm for sensor- and communication-enabled automated vehicles," in *2015 IEEE Intelligent Vehicles Symposium (IV)*, 2015, pp. 1328-1333.

[3] P. G. Gipps, "A model for the structure of lane-changing decisions," *Transportation Research Part B Methodological,* vol. 20, no. 5, pp. 403-414, 1986.

[4] A. Kesting, M. Treiber, and D. Helbing, "General Lane-Changing Model MOBIL for Car-Following Models," *Transportation Research Record,* vol. 1999, no. 1, pp. 86-94, 2007/01/01 2007.

[5] X. Li, X. Li, X. Yao, and B. Jia, "Modeling mechanical restriction differences between car and heavy truck in two-lane cellular automata traffic flow model," *Physica A Statistical Mechanics & Its Applications,* vol. 451, p. S0378437116000479, 2016.

[6] F. Meng, J. Su, C. Liu, and W. Chen, "Dynamic decision making in lane change: Game theory with receding horizon," in *2016 UKACC 11th International Conference on Control (CONTROL)*, 2016, pp. 1-6.

[7] Z. N. Sunberg, C. J. Ho, and M. J. Kochenderfer, "The value of inferring the internal state of traffic participants for autonomous freeway driving," in *2017 American Control Conference (ACC)*, 2017, pp. 3004-3010.

[8] O. Scheel, L. Schwarz, N. Navab, and F. Tombari, "Situation Assessment for Planning Lane Changes: Combining Recurrent Models and Prediction," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, 2018, pp. 2082-2088.

[9] S. Jeong, J. Kim, S. Kim, and J. Min, "End-to-end learning of image based lane-change decision," in *2017 IEEE Intelligent Vehicles Symposium (IV)*, 2017, pp. 1602-1607.

[10] M. Kuderer, S. Gulati, and W. Burgard, "Learning driving styles for autonomous vehicles from demonstration," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 2641-2646.

[11] M. Kaushik, V. Prasad, K. M. Krishna, and B. Ravindran, "Overtaking Maneuvers in Simulated Highway Driving using Deep Reinforcement Learning," in *2018 IEEE Intelligent Vehicles Symposium (IV)*, 2018.

[12] M. Zhu, Y. Wang, J. Hu, X. Wang, and R. Ke, "Safe, Efficient, and Comfortable Velocity Control based on Reinforcement Learning for Autonomous Driving," *arXiv preprint arXiv:1902.00089,* 2019.

[13] C. Hoel, K. Wolff, and L. Laine, "Automated Speed and Lane Change Decision Making using Deep Reinforcement Learning," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, 2018, pp. 2148-2155.

[14] C. You, J. Lu, D. Filev, and P. Tsiotras, "Highway Traffic Modeling and Decision Making for Autonomous Vehicle Using Reinforcement Learning," in *2018 IEEE Intelligent Vehicles Symposium (IV)*, 2018, pp. 1227-1232.

[15] B. Mirchevska, C. Pek, M. Werling, M. Althoff, and J. Boedecker, "High-level Decision Making for Safe and Reasonable Autonomous Lane Changing using Reinforcement Learning," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, 2018, pp. 2156-2162.

[16] E. Conti, V. Madhavan, F. P. Such, J. Lehman, K. O. Stanley, and J. Clune, "Improving Exploration in Evolution Strategies for Deep Reinforcement Learning via a Population of Novelty-Seeking Agents," *arXiv:1712.06560v3,* 2017.

[17] T. Salimans, J. Ho, X. Chen, S. Sidor, and I. Sutskever, "Evolution Strategies as a Scalable Alternative to Reinforcement Learning," *arXiv: 1703.03864,* 2017.

[18] D. Adjodah, D. Calacci, Y. Leng, P. Krafft, E. Moro, and A. Pentland, "Improved Learning in Evolution Strategies via Sparser Inter-Agent Network Topologies," presented at the 31st Conference on Neural Information Processing Systems, Long Beach, CA, USA., 2017.

[19] A. Rizaldi *et al.*, "Formalising and monitoring traffic rules for autonomous vehicles in Isabelle/HOL," in *International Conference on Integrated Formal Methods*, 2017, pp. 50-66: Springer.

[20] P. G. Gipps, "A behavioural car-following model for computer simulation," *Transportation Research Part B: Methodological,* vol. 15, no. 2, pp. 105-111, 1981/04/01/ 1981.

[21] S. Shalev-Shwartz and A. Shashua, "On the Sample Complexity of End-to-end Training vs. Semantic Abstraction Training," 2016.