IEEE Access
Multidisciplinary ⁝ Rapid Review ⁝ Open Access Journal

Received November 5, 2020, accepted November 6, 2020, date of publication November 10, 2020,
date of current version December 10, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3037129

A Novel Approach to Coordinating Green Wave System With Adaptation Evolutionary Strategy

YE ZHENG[1], RENZHONG GUO[1], DING MA[1], ZHIGANG ZHAO[1,2], AND XIAOMING LI[1,3]

[1]School of Architecture and Urban Planning, Research Institute for Smart Cities, Shenzhen University, Shenzhen 518061, China
[2]Key Laboratory of Urban Land Resources Monitoring and Simulation, Shenzhen University, Shenzhen 518061, China
[3]Guangdong Provincial Laboratory of Artificial Intelligence and Digital Economy (Shenzhen), Shenzhen University, Shenzhen 518061, China

Corresponding author: Zhigang Zhao (zhaozgrisc@szu.edu.cn)

This work was supported in part by the National Key Research and Development Program of China under Grant 2019YFB210310 and Grant 2019YFB2103104, in part by the China Postdoctoral Science Foundation under Grant 2019M663070, in part by the National Natural Science Foundation of China under Grant 4197134, and in part by the Open Fund of the Key Laboratory of Urban Land Resources Monitoring and Simulation, Ministry of Land and Resources, under Grant KF-2018-03-036.

ABSTRACT Urban arterial traffic coordination control has attracted much attention in smart city construction process. To achieve optimal signal timing, many studies have attempted to adjust green splits of a cycle time according to the distance between road intersections. However, existing green wave traffic control systems usually require a sophisticated calculation that depend upon the stability of vehicle speed and traffic flow, which can lead to weak robustness. Therefore, this article proposes two novel approaches to control arterial traffic coordination with the help of artificial intelligence: DDPG-BAND and ES-BAND. DDPG-BAND has two stages: a coarse-tuning stage reduces the blocking coefficient, and a fine-tuning stage optimizes the traffic evaluation index. ES-BAND introduces the Covariance Matrix Adaptation Evolutionary Strategy (CMA-ES), a scalable alternative to reinforcement learning, into signal timing. Different traffic variables are adopted as parameters to search for the optimal value by the CMA-ES. To evaluate the feasibility and effectiveness of our approaches, we import real traffic flow data from Zhongshan Road, Ningbo, Zhejiang Province, China, into a traffic environment simulator for training and then conduct a series of experiments. The results show that ES-BAND outperforms the traditional methods in terms of better convergence, lower journey time, fewer stops, and more throughput.

INDEX TERMS Green wave traffic, artificial intelligence, optimization, signal control, smart city.

I. INTRODUCTION

Traffic congestion is among the most challenging problems in urban management, especially as the car ownership rates increase in most Chinese cities. Urban arterial traffic coordination usually accounts for most of a city's traffic volume and contributes significantly to alleviating traffic pressure. The green wave system, which has become a trending feature in smart city construction process, plays an essential role in an intelligent transportation system. The green wave system maximizes the number of green lights to be passed along a road when vehicles pass the first light. Therefore, the green wave system can reduce the average stops that vehicles make and thereby improve the through efficiency of road networks [1], [2].

The traditional green wave method usually obtains the maximum bandwidth of the object function either mathematically or graphically according to the distance between traffic signal roads and the green wave velocity [3]–[9]. However, the existing approaches have two major limitations. The first limitation is similar velocities for all vehicles; vehicles (even if only a minority) whose speeds are inconsistent with the green wave velocity break the order of the entire green wave queue. The second limitation is steady traffic flow; the randomness of traffic flow changes the split and the offset, which impairs the robustness of the green wave system.

In recent years, the rapid development of state-of-the-art technologies in artificial intelligence, such as fuzzy logic [10], [11], genetic algorithms [12] and expert system [13], has introduced new concepts into traffic signal control. One example is the application of deep reinforcement learning [14]. After defining the step actions [15]–[18], reward function [19], and performance

The associate editor coordinating the review of this manuscript and approving it for publication was Gustavo Olague.

metrics, a reinforcement learning algorithm can seek the optimal timing in a corresponding traffic environment. Although remarkable achievements have been made in the study of traffic trunks, these methods do not consider the variety of real-time speeds for different types of vehicles, such as large trucks, minicars, and buses, and the sparse reward of a traffic environment in reinforcement learning makes training difficult to converge.

Covariance Matrix Adaptation Evolutionary Strategy (CMA-ES) is an alternative to popular MDP-based RL techniques such as Q-learning and policy gradients [20]. Unlike the policy gradients in reinforcement learning, a CMA-ES replaces back-propagation with smoothing of the cost function in parameter space, which can reduce the amount of computation per episode by approximately two-thirds, and memory requirements by potentially much more. For this reason, in the present article, we implement a reinforcement solution for traffic signal control. We elaborate the training steps and explicitly define the observation, action, and reward of the reinforcement learning application. We thus propose the ES-BAND (Evolutionary Strategy-BAND), a novel approach that introduces the CMA-ES for signal timing training data to coordinate the traffic signals on an arterial. ES-BAND converts the signal timing at each intersection into multiple search feature vectors and then finds the optimal value under the objective function based on the evolutionary strategy. ES-BAND can avoid traffic cases with long action sequences and delayed rewards and can easily perform parallel processing for performance improvement. The proposed approach has been put into practice with the help of the Ningbo Traffic Bureau, and the solution has been proven useful.

The contributions of the ES-BAND, compared with its previous version, are summarized as follows:

1. We consider the robustness of traffic signal coordination under different vehicle speeds and departure rules. Both AI solutions use an open-sourced simulator which can not only change vehicle speed, but also can simulate different departure rules of the same traffic volume. Therefore, the robustness of the algorithm is enhanced because various traffic environments are compatible with the optimization of timing schemes.
2. ES-BAND uses the CMA-ES to find the optimal value without a requiring substantial effort to design the reward function.
3. In ES-BAND, a distributed computing approach is provided to accelerate training.

The remainder of the article is organized as follows. After reviewing the related work on the green wave system in Section 2, we present the specific algorithmic process of ES-BAND in Section 3. Section 4 provides a case study of coordinating the traffic signals on Zhongshan West Road, an arterial in Ningbo, China, using ES-BAND. Finally, Section 4 concludes the study and outlines potential applications for this research.

II. LITERATURE REVIEW

A. BANDWIDTH-BASED OPTIMIZATION STRATEGY

Maximizing the green wave band is the most commonly used traditional objective for arterial traffic signal optimization. The literature shows that stop-and-go behaviors, journey time, and throughput can be reduced by a width-based optimization strategy. Morgan *et al.* found that the band-width in each direction is generally single. Under this condition, they developed an algorithm for solving two kinds of problems related to synchronizing traffic signals for a progression on an arterial street [3]. Little *et al.* formulated a mixed-integer line program to solve more general problems [4]. MAXBAND [9], which is a portable, off-line, FORTAN computer program, was developed by Little *et al.* for setting arteries to achieve maximal bandwidth. MAXBAND also has several newer versions – MAXBAND-96 [20] and MAXBAND-3.1 [21] – to improve performance. Yang *et al.* presented a multipath bandwidth-based model that can provide green wave for multiple paths on main roads [22]. Tsay and Lin used a general mixed-integer programming formulation, based on which a program called BandTop was developed to obtain the real progression bandwidth [23]. However, the band in BandTop only increases in width along the arterial, meaning it could not adequately adapt the bandwidth to variations of flow. MultiBAND [8], [24] generate variable bandwidth progression schemes in which each directional road section is assigned an individually weighted band, which offers the traffic engineer a much wider range of design options than existing arterial progression methods do. Aiming to restrict symmetry in MultiBAND, Zhang *et al.* proposed an asymmetrical multiband model called the AM-Band for arterial traffic signal coordination [25]. To address an increasing number of signals, Tian and Urbanik [26] divided a large signalized arterial into the subsystem and each subsystem was optimized to achieve the maximum throughput. Recently, scholars have improved the MAXBAND algorithm by adjusting the priority of specific vehicles, such as trams [27] and buses [28]. Considering the uncertainty of progression time, Li [29] develop a two-phase approach. They generated a number of optimal or suboptimal plans by perturbation controlled MAXBAND in the first phase and simulate random progression time to evaluate these candidate solutions in the second phase.

In summary, a width-based optimization strategy requires all directions to have the same (or an integer multiple) cycle. Considering the green split, these approaches adjust the offset of each intersection to obtain the maximal green wave bandwidth.

B. ARTIFICIAL INTELLIGENCE-BASED OPTIMIZATION STRATEGY

In addition to traditional bandwidth-based solutions, artificial intelligence methods have been applied successfully to the traffic control problem. Evolution Strategy [20] (ES) is a class of black-box optimization algorithms that has been extremely

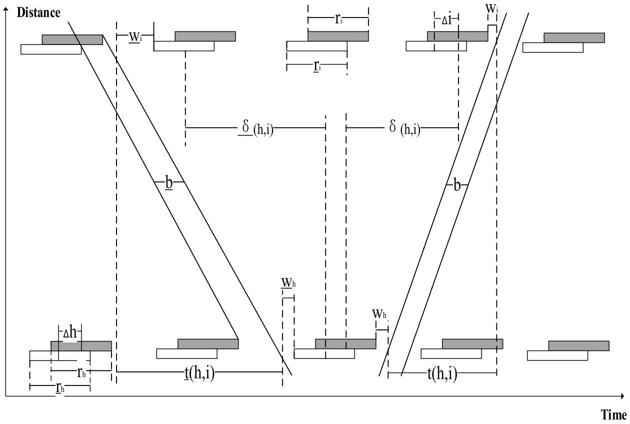

FIGURE 1. **Illustration of MAXBAND computing process: a traditional way to coordinate arterial traffic by formulating a mixed-integer linear program.**

successful in solving optimization transport problems in low to medium dimensions [31], [32]. As the evolution in real world, the best population is chosen for each generation to form the next generation until the objective is fully optimized. The CMA-ES is a special example of evolution strategy when a population is represented by a full-covariance multivariate Gaussian.

Regarding the control of consecutive intersections on artery, few studies have introduced the artificial intelligence method of achieving the green wave effect. Kong *et al.* presented a two-direction green wave intelligent control strategy that includes a coordination layer and a control layer [33]. This control strategy can maximize the possibility for vehicles in each direction along an arterial road to pass the local intersection without stopping when the utility efficiency of the green signal time is at a relatively high level. Ma *et al.* introduced an intelligent technique based on an adaptive genetic-artificial fish swarm algorithm to optimize a green wave traffic control system [34]. They also applied this technology to the arterial road in Lanzhou, China and achieved good results.

III. PROBLEM SETTING

This section describes the preconditions and optimization objective of this article. In our application, the traffic signal executes the control by using the following preconditions:

(1) Our application changes the green light time at each phase (green split), cycle, and offset for signal timing.
(2) The changeable phase sequence is not considered.
(3) The yellow-light time is fixed in our application.

The optimization objectives are different under different situations. Normally, the average number of vehicles stop-and-go behaviors is the first consideration in the green wave coordination system. However, during rush-hours or off-peak hours when the traffic flow is particularly intense or free-flowing, the vehicles' throughput and journey time should be considered. In the following section, the average number of vehicles' stop-and-go behaviors, throughput, and journey time are collectively referred to as the **Traffic Evaluation Index**. In sum, the optimization objective in our system consists of two parts. The top priority is no congestion in the traffic system, and the secondary priority is the optimization of the traffic evaluation index.

IV. DDPG-BAND

A. OVERVIEW OF REINFORCEMENT LEARNING

Reinforcement learning focuses on the trial-and-error interaction of goal-directed agents with a dynamic environment, in which an optimal action sequence with maximum cumulative reward is learned [35]. The fundamental mathematical model of RL is the Markov decision process (MDP), which examines a sequential decision-making task. An agent that uses the MDP consists of a set of states (also called observations) $\mathcal{S}$, a set of actions $\mathcal{A}$, a reward function $\mathcal{R}$ [36], a state transition function $\mathcal{P}$, and a discount rate Y (Y $\in$ [0, 1]), which is denoted as Eq. (1):

$$\mathrm{M} = \langle \mathcal{S}, \mathcal{A}, \mathcal{R}, \mathcal{P}, \Upsilon \rangle \tag{1}$$

The policy function takes a state and an action as parameters and returns the probability of taking the action in that state.

If π means the ***action policy*** where an agent chooses his action based at each time step, for many Markov decisions, the optimal strategy is determined; that is Eq. (2):

$$\pi_{max}(s) = a \tag{2}$$

The state value functionis one of the performance criteria used to evaluate the policy function. In every decision of a state process, an agent has a set of actions selected as a set of alternative actions and each action leads to different rewards. If the reward value changes from state s_t to state s_{t+1} as Eq. (3):

$$r_t = \mathcal{P}(s_i, \pi(s_i)) \tag{3}$$

then the sum of these sequential rewards R_t can be derived as Eq. (4):

$$R_t = \sum\nolimits_0^\infty \Upsilon^k r_{t+k+1} \tag{4}$$

On the above basis, the state value function is defined as Eq. (5):

$$V^\pi(s) = E_\pi(R_t | \mathcal{S}_t = s, \mathcal{A}_t = a) \tag{5}$$

The agent in a Markov decision process tries to seek the optimal policy π, which maximizes the expected accumulative maximum rewards starting from any state $\mathcal{S}'$; that is, we should find the best solution to the state value function written as $q_*(s, a) = max(V^\pi(s))$. According to the Bellman Equation [37], the state value function of any state-action pair (s, a) under any policy π can be approximated iteratively as Eq. (6):

$$q_*(s, a) = \sum\nolimits_{s',r} \mathcal{P}(s', r | s, a)\left(r + \Upsilon \max\left(q_*(s', a')\right)\right) \tag{6}$$

B. REINFORCEMENT LEARNING IN TRAFFIC CONTROL

There has been renewed interest in traffic light control with the advent of the deep reinforcement learning (DRL) approach [38]–[40], but most related studies have not provided implementation details. Reinforcement learning (RL) is a primary approach to learn control strategies by considering what actions autonomous agents should take to maximize a numerical reward signal [41]. Since Abdulhai *et al.* [42] proposed the first truly adaptive traffic signal that learns to control the traffic signal dynamically, many studies have shown the potential capacity of RL in traffic control. Mannion *et al.* [14] presented a comprehensive review of previous literatures (before 2016) on the RL of the traffic control problem. The review summarized, in detail, how these approaches were defined using the action, state and reward function. With the development of smart cities, related research has focused on using novel technologies such as edge computing and 5G effectively to help highlight the advantages of RL in traffic control. Ning *et al.* [43] developed an intent-based traffic control system by investigating DRL for 5G-envisioned IoCVs, which can dynamically orchestrate edge computing and content caching. Zhou *et al.* [44] proposed a solution based on edge computing nodes to collect traffic data. ERL alleviates congestion. Joo *et al.* [45] designed a TSC system, from the perspective of smart city construction, to maximize the number of vehicles crossing an intersection and balance the signals between roads by using Q-learning (QL). For the current situation in which most studies do not consider realistic settings, Tan *et al.* [46] proposed a DRL-based adaptive traffic signal control framework that explicitly considers realistic traffic scenarios, sensors, and physical constraints. Wang *et al.* [47] proposed a cooperative double Q-learning to coordinate large-scale traffic signal control. Liang *et al.* [48] applied RL to control a traffic light cycle. In their methods, the state of RL is traffic data divided into grids, the actions are the duration changes of a traffic light, and the reward is the cumulative waiting time difference between two cycles. To address a problem where only a few vehicles are equipped with wireless communications capability, Cabrejas-Egea *et al.* [49] compared the performance of agents using different reward functions in a simulation across various demand profiles and subject to real world constraints. Zhang *et al.* [50] reported a new RL algorithm for partially detected intelligent traffic signal control (PD-ITSC) systems, which can perform well under a small detection rate environment. Xiong *et al.* [51] proposed a novel method to leverage demonstrations collected from classic methods to accelerate learning, which is mainly based on the state-of-the-art deep RL method Advantage Actor-Critic (A2C).

Traditional reinforcement learning uses a two-dimensional table (Q-Table) to store the state and corresponding action of the agent in every step, which cannot deal with huge state spaces and continuous candidate action in real-world application. With the development of deep learning in recent years, Deep Q-Network [52] (DQN) have been made proposals for how to replace Q-Table with deep neural network to solve this problem. Deep Deterministic Policy Gradient (DDPG) [53] is an improvement on DQN to enable it to support continuous action. DDPG is one kind of A2C algorithms, where the actor network can choose the next action based on probability and the critic network evaluates the actor's behavior and updates the above probability. DDPG-BAND (Deep Deterministic Policy Gradient-BAND) is applied to DDPG as the learning policies in our reinforcement solutions. There are two stages in the training process. The first stage is the coarse-tuning stage and its objective is to train the agent controlling the traffic signal light to avoid road congestion. In the coarse-tuning stage, we train the agent in each intersection separately because the massive dimensions of action and observation make the algorithm difficult to converge when the entire arterial road is taken as a single agent. When the coarse-tuning stage is completed, we combine the model of each intersection into the main road coordination agent for fine-tuning training. The first and second stages have similar observations, while the only difference lies in the number of dimensions. We will elaborate the state, action, and reward of DDPG-BAND in the following section.

C. STATE DESCRIPTION

In the case of traffic systems, the state of the RL model should reflect the traffic situation of intersections. ***Vehicle Queuing Quantity*** means the total number of vehicles in line that are waiting for a red light to turn green. At a certain intersection, the vehicle queuing quantity in each direction is a benchmark for the rate of traffic flow in that direction, which directly determines the green light time of the phase. However, it is not insufficient to simply count the number of vehicles and not care about the vehicle type. The acceleration rate of a large truck is far lower than that of an ordinary car, so more green time is required. For this reason, a certain weight should be added to the vehicle queuing quantity according to the vehicle characteristics, and the ***vehicle queuing number*** is thus expressed as the weighted quantity.

Based on the above conclusion, there are two granularities of state space in our system. The state space with a coarse grain size only approximates the vehicle queuing quantity of edges in each direction, while the fine-grained state space provides exact statistics for each lane. For example, Figure 2 illustrates an intersection with four directions. The state space with coarse grain size for this intersection can be expressed by a vector of four dimensions (the outgoing lanes are not included). Each element of the four-dimensional vector represents the vehicle queuing number in each direction, including east, south, west and north. The fine-grained state would have to map the Figure 2 intersection to an eight-dimensional vector where the vehicle queuing numbers in each lane are stored.

D. ACTION SPACE

After the agent has observed the state of the environment, it must choose one action from the set of all available actions. Traffic lights should take corresponding measures when facing different traffic conditions

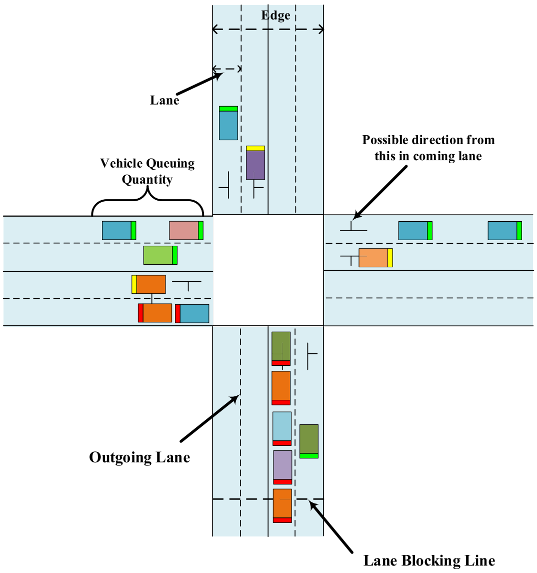

FIGURE 2. Illustration of traffic crossroads. Different color at the edge of vehicles represents their taillights. Red means vehicles are stopping, yellow means vehicles are slowing down and green means vehicles are running.

In the coarse-tuning stage, the action candidates are the preliminary green times of each traffic signal phase. When the traffic flow is relatively high during the rush hour, a longer green time increases throughput to reduce traffic jams. On the other hand, when traffic flow reaches a free-flow state during off-peak hours, less green time is required. Therefore, the action space is a multidimensional vector, where each dimension of the vector represents the green time of each phase in the traffic signal. Therefore, the range of action vectors in the coarse-tuning stage should consider the following factors:

- The preliminary green time is a positive integer.
- The preliminary green time must have a fixed minimum that is determined by the length of the road to ensure that pedestrians can pass.
- The complete cycle of the traffic signal must not exceed a certain value determined by the longest waiting time that can be endured.

In the fine-tuning stage, the entire road is taken as a single agent. We first combine the preliminary green time of each intersection and, on this basis, we adjust the green time to a certain extent to improve the traffic efficiency of arterial roads on the premise of no traffic jam. After the agent takes an action, the traffic simulator can obtain the final green time for each phase at each intersection and run for a certain but possibly unequal simulation time. (For example, the agent can take an integral multiple of cycles as the simulation time, and the traffic light can have different cycles with different actions.) Finally, the agent evaluates this behavior according to the reward function and takes the next action. How the reward function is defined will be discussed in the next section.

E. REWARD FUNCTION

The reward function is the final element of RL. After the agent has observed the state of the environment and performed an action, it receives the reward from the reward function in the environment. The reward function can ensure that the agent finally takes an action based on our optimization objectives. The reward function in the coarse-tuning stage is different from that in the fine-tuning stage.

We first discuss the details of the coarse-tuning reward function. The objective of the coarse-tuning stage is to reduce the blocking coefficient for each intersection. As shown in Figure 2, the ***lane block line*** is a threshold line of the maximum vehicle queue length of the road that does not cause congestion. In practice, the ***lane block line*** can be set to a distance from the end of the intersection. For example, the distance from the blocking line is set to the end of the intersection that occupies 10% of the length of the road. Such a location for distance setting can be easily obtained from remote sensing images or by fieldwork. When the signals are used for traffic control, the vehicle queue length of each lane must not exceed the lane block line. ***The blocking coefficient*** denotes how many times the queuing length exceeds the lane block line during the simulation. Let the ***blocking coefficient ratio*** denote the ratio of the variable blocking coefficient to the simulation time. Based on the above conditions, the reward function should be considered from the following situations. If after the simulation, the variable blocking coefficient ratio ranges from zero to nonzero, then it returns directly to -1; if it is from zero to nonzero, it returns to 1. If both of the blocking coefficient ratios in two actions are greater than or equal to zero, the reward function returns to 0.

In the fine-tuning stage of DDPG-BAND, we optimize the traffic evaluation index of the whole artery road. During the simulation time, the traffic evaluation index can normally be obtained from the simulator. For example, we put speed detectors on every vehicle of the simulation, therefore, the stop-and-go behavior is calculated as follows:

1. If the vehicle comes to a complete stop, that is, the speed is zero, we add stop-and-go behavior twice.
2. If the vehicle has braking behavior (acceleration less than 0) and the speed has dropped to a certain extent, 5 km/h for example, we add stop-and-go behavior once.
3. If the vehicle stops and goes, the stop-and-go behavior will continue to accumulate.

However, the traffic evaluation index is likely to be delayed during the process. It takes a certain amount of time for the vehicles affected by the current action to run through the traffic scene. Therefore, we insert the traffic evaluation indexes after each simulation into a traffic evaluation FIFO queue with a certain length. After each simulation, we average the data in the traffic evaluation FIFO queue. If the average value improves, it returns to 1. If the average value become worsens but remains within a certain tolerance, it returns to 0; otherwise, it returns to -1.

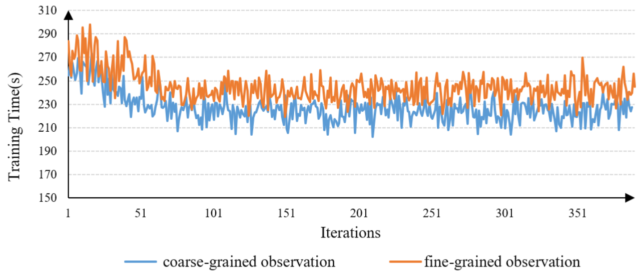

FIGURE 3. Time consumed in each iteration for two granularity observations on Jiefang road. The experimental data are described in Section 4; the coarse-grained observation is a four-dimensional vector and the fine-grained observation is an 18-dimensional vector.

F. DISCUSSION ON DDPG-BAND

Through the above process, we implement a traffic control method based on RL. RL can be a potential solution for traffic control. However, DDPG-BAND also has drawbacks.

The first disadvantage is that the simulation time is difficult to define. In our approach, the agent obtains the observation before the simulation and computes the reward function afterwards; as a consequence, the long simulation time may lead to an obsolete observation for the agent. On the other hand, the insufficient simulation time causing the action of the agent is replaced by the next action before its effect on the traffic evaluation index is calculated in the reward function. The essential reason for this contradiction is that the agent observation is not static, as it computes the reward function in DDPG-BAND.

Another weakness of DDPG-BAND is the slow convergence. There are two possible reasons for this. The first is that, for many cases in our experiment, the reward function returned to a zero value and the agent received a sparse matrix from the traffic environment. The second reason is the large search space including the observation space and the action space. The observation in our approach contains two granularities. As shown in Figure 3, the average time spent on coarse-grained observation is 8 percent longer than that of the fine-grained (227 s versus 245 s).

Even under the coarse-grained observation, each intersection has four dimensions (directions). For the artery, the total observation space will reach 10 or even dozens of dimensions. The action space is similar. When multiple traffic signals are controlled by a single agent, the action space is the sum of all their phases on the artery, resulting in a high dimension for the action space. Figure 4 shows the convergence of the blocking coefficient with two granularity observations. In Fig. 4, we can see that the convergence rate of fine-grained observation is slightly slower than that of coarse-grained but they have similar precision.

V. ES-BAND

In this section, we will introduce our ES-BAND approach, a novel framework for a green wave traffic control system that utilizes CMA-ES [29]. The problem setting is the same as DDPG-BAND represented in Section III.

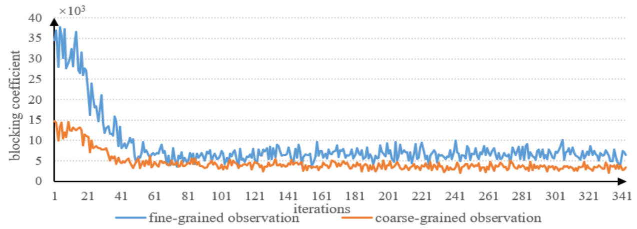

FIGURE 4. The blocking coefficient in the training of the reinforcement learning algorithm with two granularity observations on Jiefang Road.

A. OVERVIEW

The basic equation for the sampling feature vector refers to Eq. (7):

$$x_k^{(g+1)} \sim N\left(m^{(g)}, \left(\sigma^{(g)}\right)^2 C^{(g)}\right) \quad for\ k = 1, \ldots, \lambda \tag{7}$$

where

$\sim$ denotes that the left and right sides obey the same distribution.

N is the multivariate normal search distribution.

$x_k^{(g+1)}$ refers to the k$^{\text{th}}$ search feature vector from generation g + 1.

$m^{(g)}$ means the value of the search distribution at generation g.

$C^{(g)}$ is the covariance matrix at generation g.

λ is the sample size.

Each iteration of the ES-BAND algorithm mainly consists of two steps (see Fig. 5):

(1) Computing the green time for each phase generated by sampling a multivariate normal distribution and then the traffic simulator [54] returns the objective functions.

(2) Combining the objective value, calculating the stochastic gradient estimate, and updating the parameters. As with DDPG-BAND, the priority of the blocking coefficient is higher than that of the traffic evaluation indicator. In addition, the calculation of the blocking coefficient takes all lanes into account, while the traffic evaluation indicator only considers lanes in the direction of the main road.

B. DEFINITION OF FEATURE VECTOR

Since the sampling of feature vectors in CMA-ES is a random process, the same initial value produces different results after the same training. We run the process for three to five times to find the best answer. The detailed steps of the above process are described in the following paragraph.

The CMA-ES samples the independent points from a specific distribution (such as normal distribution) and then iteratively chooses the best points as the next generation. Consequently, the first step of ES-BAND is to define the search points. Studies [8], [9] have shown that an identical cycle (referred to below as the public cycle) of lights at each intersection is required to maintain the green wave effect. For each intersection, we can obtain the green time in the last

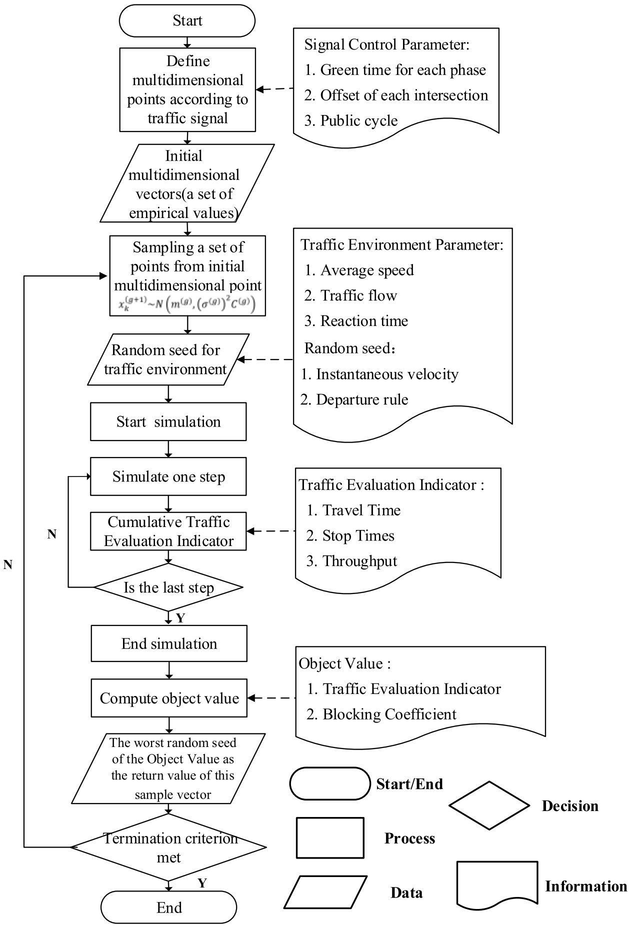
Start
Define multidimensional points according to traffic signal
Signal Control Parameter:
1. Green time for each phase
2. Offset of each intersection
3. Public cycle
Initial multidimensional vectors(a set of empirical values)
Sampling a set of points from initial multidimensional point
$x_k^{(g+1)} \sim N\left(m^{(g)}, \left(\sigma^{(g)}\right)^2 C^{(g)}\right)$
Traffic Environment Parameter:
1. Average speed
2. Traffic flow
3. Reaction time
Random seed:
1. Instantaneous velocity
2. Departure rule
Random seed for traffic environment
Start simulation
Simulate one step
Cumulative Traffic Evaluation Indicator
Traffic Evaluation Indicator :
1. Travel Time
2. Stop Times
3. Throughput
N
N
Is the last step
Y
End simulation
Compute object value
Object Value :
1. Traffic Evaluation Indicator
2. Blocking Coefficient
The worst random seed of the Object Value as the return value of this sample vector
Termination criterion met
Y
End
Start/End
Decision
Process
Data
Information

FIGURE 5. **Flow chart of the ES-BAND algorithm. Note: In the sampling process, different random seeds are considered that improve the robustness for dealing with various traffic environments.**

phase by subtracting the sum of other phase times from the public cycle.

Suppose intersection C_k has P_k phases and we have $P_k - 1$ dimensions for the green time needed to set at each intersection. In addition, each intersection, except for the first one, has an offset time plus the public cycle, so the total number of dimensions of search points is as in Eq. (8) below:

$$\sum_{1}^{k=n} (C_k - 1) \times P_k + (n-1) + 1 \tag{8}$$

Simplified as Eq. (9)

$$\sum_{1}^{n} k\, C_k \times P_k \tag{9}$$

C. TRAFFIC SIMULATION

Sampling from a normal distribution can generate a set of sample points, which are taken as the green time of each phase for the simulation. Ultimately, a set of traffic evaluation indexes is obtained as return values. During the simulation, it is possible to generate traffic congestion in the traffic environment. For example, the simulator fails to start normally when one of the lanes at a crossroads is full; in this case, we give the worst return value. Note that this step can be performed in a multi-threaded or distributed manner to reduce the training time.

D. OFFSPRING

A set of function values obtained through simulation are used as the next generation of CMA-ES. The generated offspring sample points are processed using the following steps:

(1) The offset of the next generation can be added to or subtracted from the public cycle to keep the value between zero and the public cycle.

(2) The last green light time is recalled by subtracting the sum of other phase timings from the public cycle. Consequently, if the last green light time is less than the minimum green light time, it is necessary to adjust the public cycle, which means that the last green time at all sections increases accordingly.

E. OVER-FITTING

The corresponding timing scheme can be obtained after the convergence. However, this timing scheme has a robust correlation with departure rules and it is difficult to guarantee that our simulator's departure rule is same as that in reality. Therefore, we need to find a timing scheme that can adapt to various environments and has better generalization ability to solve the overfitting problem.

To do this, we rely on random factors that commonly appear in most simulators. Under the same departure probability, different random seeds of the simulator produce different departure rules. Therefore, multiple groups of random seeds are used in the simulator and the worst result is taken as the return value. The above process can also be executed in parallel.

F. DEFINITION OF OBJECTIVE FUNCTION

The definition of object value f_x consists of the following two parts:

- **Blocking Coefficient:**During training, all lanes should not be blocked. Assume the blocking coefficient is α. Computing the occupancy ratios for all lanes in each iteration; if the waiting vehicles have exceeded the lane block line, then $\alpha = \alpha + 1$.
- **Traffic Evaluation Index:**The training should stress minimizing a specific traffic evaluation index.

As with DDPG-BAND, the priority of the blocking coefficient is higher than the traffic evaluation indicator. In addition, the calculation of the blocking coefficient takes all lanes into account, while the traffic evaluation indicator only considers lanes in the direction of the main road.

G. THE OPTIMAL SOLUTION

ES-BAND generates multiple traffic environments by random seeds to increase the robustness of the algorithm. Three groups of 10 random seeds were conducted as training

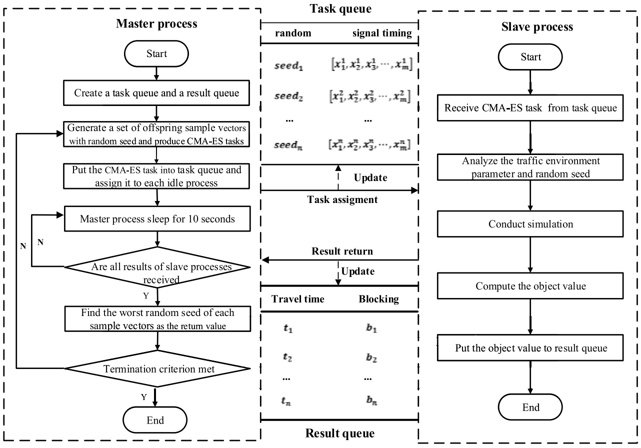
Master process
Start
Create a task queue and a result queue
Generate a set of offspring sample vectors with random seed and produce CMA-ES tasks
Put the CMA-ES task into task queue and assign it to each idle process
Master process sleep for 10 seconds
N
N
Are all results of slave processes received
Y
Find the worst random seed of each sample vectors as the return value
Termination criterion met
Y
End
Task queue
random
signal timing
$seed_1$
$[x_1^1, x_2^1, x_3^1, \cdots, x_m^1]$
$seed_2$
$[x_1^2, x_2^2, x_3^2, \cdots, x_m^2]$
...
...
$seed_n$
$[x_1^n, x_2^n, x_3^n, \cdots, x_m^n]$
Update
Task assigment
Result return
Update
Travel time
Blocking
t_1
b_1
t_2
b_2
...
...
t_n
b_n
Result queue
Slave process
Start
Receive CMA-ES task from task queue
Analyze the traffic environment parameter and random seed
Conduct simulation
Compute the object value
Put the object value to result queue
End

FIGURE 6. Task scheduling process.

samples and another three groups of equal numbers of random seeds were the testing samples. Finally, the optimal solution of this algorithm is the timing with the lowest mean of objective function.

H. PARALLEL COMPUTING

ES-BAND has triple loops that can be executed in parallel: (1) ES-BAND samples λ offspring as green times for each phase of the traffic lights at each intersection; (2) every offspring has several random seeds; (3) we run the whole algorithm several times for the randomness of ES-BAND, with each attempt being independent of each other, which can be processed in parallel. The first two triple loops are conducted at each iteration of ES-BAND.

As shown in Fig.6, the simulation process of ES-BAND can be divided into two categories: the master process, which collects the results of all slave nodes and samples the feature vectors by CMA-ES, and the slave executor, which conducts traffic simulation training with the corresponding timing scheme and random seeds assigned by the master executor. Each group of timing schemes combined with random seeds is allocated to the slave process for traffic simulation.

1. **Master:**A population of new samples is generated by sampling multivariant normal distribution, considered the green time at each phase. After each group of timing schemes combined with random seeds is allocated to the slave, the master process sleeps.
2. **Slave**: The traffic simulation is conducted according to the timing scheme and random seed and returns f_x to the master.
3. **Master:** After receiving f_x from all slave processes, the master process is awakened. For a particular timing scheme, the master process finds the worst-performing random seed (expressed as worst $(seed_i)$, $i \in \ 1, 2, \ldots, n$), under which f_x is obtained for the next generation.

VI. CASE STUDY

For case studies, the green wave traffic control system was tested at four consecutive intersections along Zhongshan Road in Ningbo, China. The feasibility of the ES-BAND algorithm could also be verified using the above cases. Our case studies used Simulation of Urban Mobility (SUMO) [55] as the traffic simulator, where the acceleration and deceleration [56], [57] and the reaction time [58] are also defined.

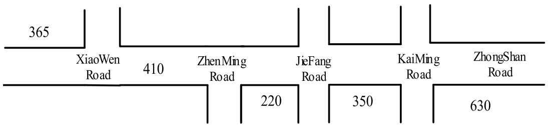

FIGURE 7. The layout of Zhongshan Road in Ningbo, China. Zhongshan Road contains four intersections: Xiaowen Road (T-junction), Zhenming Road (T-junction), Jiefang Road, and Kaiming Road.

A. ROAD SELECTION

Zhongshan Road is a main urban arterial in Ningbo City, Zhejiang Province, China, which stretches 20.2 km east-west throughout the whole city. For this study, the busiest section of Zhongshan Road was selected, whose range is from Xiaowen Road in the east to Kaiming Road in the west. The total length of the selected road is more than 1 km. The general situation of the intersection and the corresponding road network are shown in Figure 7.

B. TRAFFIC DATA

We obtained the original traffic flow data from cameras at the intersections. The traffic flow was categorized into represents large vehicles and small vehicles. Large vehicles are those with a length exceeding 10 meters, such as large trucks and buses, whereas small vehicles are small and medium-sized cars. To collect the traffic flow data every 5 minutes between 7:00 a.m. and 11:30 a.m., we performed simple processing to obtain the departure probability of each direction into the simulator.

Finally, as shown in Table 2, we obtain a departure matrix with dimensions of the number of intersections × number of crosses (3 for T-junction and 4 for crossroads) × 3 (left, right and straight).

C. IMPLEMENTATION

In the entire experiment we implemented three signal control algorithms. In addition to the above two artificial intelligence algorithms, we also took the traditional mixed-integer linear algorithms of MAXBAND [6] as the experimental benchmark. The mixed-integer linear equations are solved by the LINGO software. In DDPG-Band, coarse-grained observation is executed as the learning policies in the training.

Our application is deployed on the operating system CentOS 7. The hardware facilities in our experiment are composed of three computing nodes. Each computing node has 24 cores and 32 G RAM. Therefore, there are a maximum of 72 processes carrying out the traffic simulation at the same time. The distributed process is coupled to the QueueManager data structure. The ES-Method algorithm is then implemented using Python 3.7.3 with the covariance matrix adaptation built on Tensorflow1.14. With the

TABLE 1. The rate of traffic flow in the intersection.

Intersection	Entrance of intersection	Direction	Flow Rate (Truck/Car, Unit: vph)
Xiao Wen Road & Zhong Shan Road	East	right	266/171
		straight	369/924
	West	left	31/137
		straight	280/1130
	North	left	110/328
		right	95/127
Zhen Ming Road & Zhong Shan Road	East	straight	329/931
	South	left	223/448
		right	301/84
	West	right	190/228
		straight	235/1145
Jie Fang Road & Zhong Shan Road	East	left	472/250
		right	600/71
		straight	313/503
	South	left	373/290
		right	448/101
		straight	273/450
	West	left	114/491
		right	773/197
		straight	215/614
	North	left	299/137
		right	81/168
		straight	164/622
Kai Ming Road & Zhongshan Road	East	left	67/194
		right	237/420
		straight	475/840
	South	left	290/76
		right	796/69
		straight	73/208
	West	left	8/92
		right	6/128
		straight	348/600
	North	left	30/240
		right	231/41
		left	67/194

abovementioned configurations, the traffic simulation runs under the SUMO 1.3.1 environment (Institute of Transportation Systems, German Aerospace Center).

In our experiment, there are two types of vehicles (truck and car), and each type has different configurations in terms of length, max speed, acceleration, brake deceleration and driver reaction time. All these configurations are set according to the real-world statistics from the literature [59]. Table 2 describes the details of the configurations in simulator:

TABLE 2. configurations in the simulator.

Items	Truck	Car
Length(m)	12	4.8
Max Speed(km/h)	40-50	50-60
Acceleration(m/s^2)	0.24-0.96	0.55-1.89
Deceleration(m/s^2)	0.52-0.88	2.42-3.36
Driver Reaction(s)	0.8-1.0	0.8-1.0

D. TRAINING RESULTS OF DDPG

This section describes the learning speed results for two artificial intelligence solutions. In DDPG-BAND, we train the blocking coefficient of each intersection separately in the coarse-tuning stage. Figure 8 shows that with the increase in the number of iterations, the blocking coefficient of all four intersections decrease and remain stable when the number of iterations reaches approximately 70. For example, at the intersection of Jiefang Road, the average blocking coefficient in the first 10 iterations is nearly 3.11 times that of the last 10 iterations, which indicates that RL has a certain effect on easing road congestion.

Moreover, we also explore the change law of the traffic evaluation indexes in the fine-tuning stage.

From the above experiments, we find that the traffic evaluation indexes have been optimized. The details of the training results are as follows:

1. The average journey time of the last 20 iterations is 7.87% shorter than that of the first 20 iterations. The average journey time finally converges to approximately 145 seconds.
2. The average number of stops for the last 20 iterations is 41.37% less than that for the first 20 iterations. The average number of stop-and-go behaviors finally converges to approximately3.72.
3. The throughput of the last 20 iterations is 5.28% higher than that of the first 20 iterations. The throughput finally converges to approximately 15,390.

E. TRAINING RESULTS OF ES-BAND

Figure 12 demonstrates a similar conclusion for ES-BAND. The blocking coefficient decreases significantly until it reaches a stable value, while ES-BAND requires fewer iterations (approximately 25) than DDPG-BAND.

However, based solely on the above experimental results, it cannot be concluded that the learning speed of RL is slower than that of ES-BAND because each iteration in the two approaches takes a different amount of time. Our experiments have proven that each iteration of DDPG-BAND (JieFang Road) takes approximately 2.2 times that of ES-BAND. Figure 13 shows the learning time for both AI algorithms by weighting the experimental results in DDPG-BAND (JieFang Road) and ES-BAND. The experimental results show that ES-BAND converges slightly faster than DDPG-BAND.

We can see in Figure 14 that the traffic evaluation indexes have also been optimized after approximately 50 iterations. The details of the training results are as follows:

(1) The average journey time of the last 10 iterations is 33.10% shorter than that of the first 10 iterations.

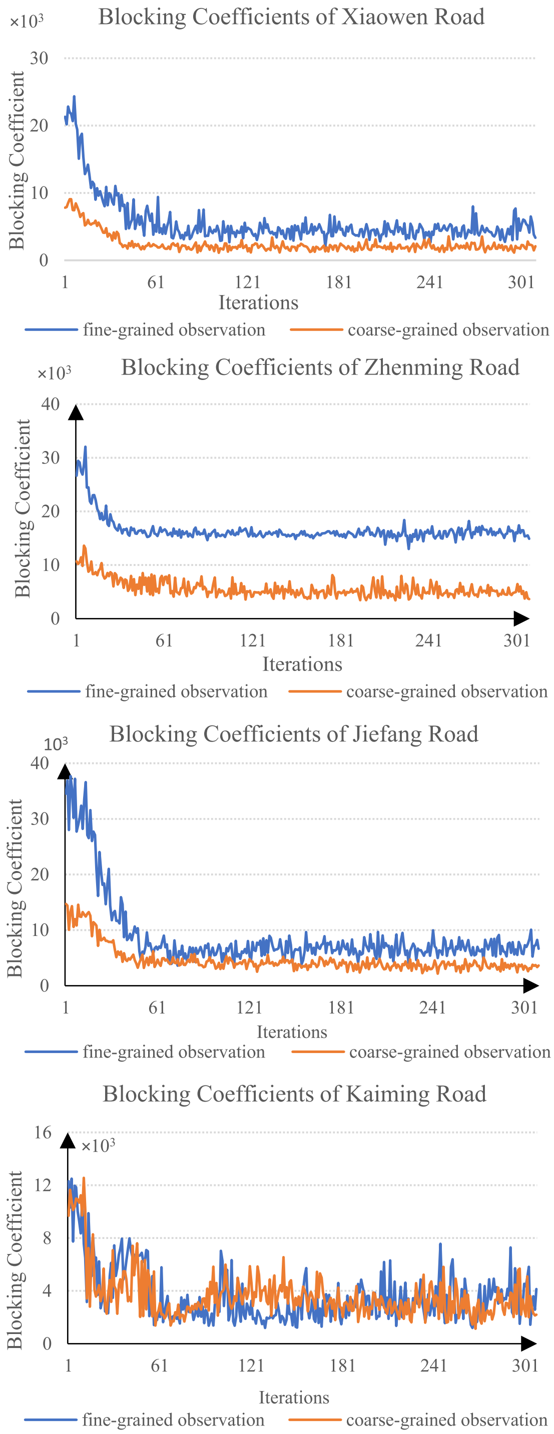

FIGURE 8. The relationship between the number of iterations and the blocking coefficient of DDPG-BAND at four intersections. The result contains two kinds of observation granularity, where the blocking coefficient of the fine-grained observation is greater than the coarse-grained observation because former accumulates the blocking coefficient of each lane, while the latter only counts once in each edge.

The average journey time finally converges to approximately 149 seconds.

(2) The average stops of the last 10 iterations is 33.12% less than that of the first 10 iterations. The average number of stop-and-go behaviors finally converges to approximately 2.80.

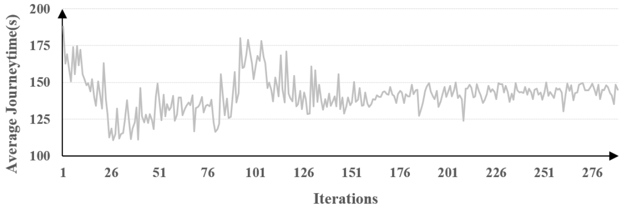

FIGURE 9. Average journey time of vehicles in each iteration. Vehicles outside the main road are not included, which is also implemented in the following experiments.

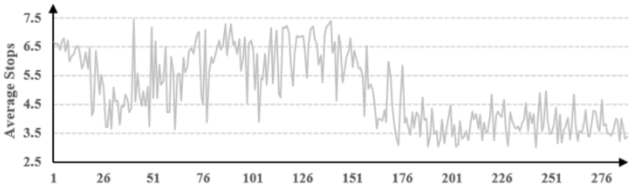

FIGURE 10. The unit of the y-axis is the average number of stop-and-go behaviors for vehicles in each iteration. If a vehicle takes stop-and-go action repeatedly when passing a single intersection, the stops will accrue one or more times correspondingly.

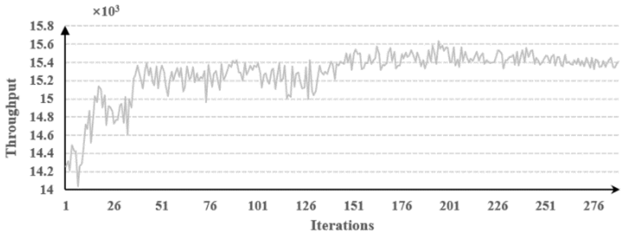

FIGURE 11. Throughput of vehicles on the arterial road. Y-axis represents the number of cars.

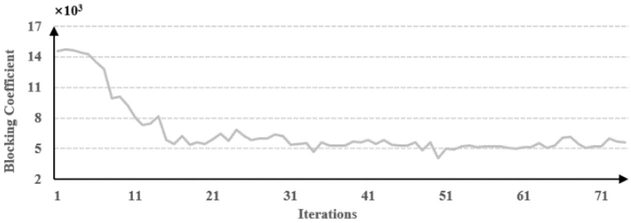

FIGURE 12. The relationship between the number of iterations and the blocking coefficient of ES-BAND on the arterial road. Unlike in the DDPG-BAND, in the ES-BAND we train the blocking coefficient of all intersection s on the arterial road as a whole.

(3) The throughput of the last 10 iterations is 11.11% higher than that of the first 10 iterations. The throughput finally converges to approximately 15,010.

In the last experiment of this section, we explored the effect of different types of parallelism on training speed. Figure 15 shows the relationship between the number of parallelisms and the average time spent per iteration.

We can see in the experimental results that, with increasing parallelism, the average training time per iteration decreases. However, when the parallelism reaches approximately 30, the average time is no longer reduced because in CMA-ES, the relationship between the number of parameters N and the

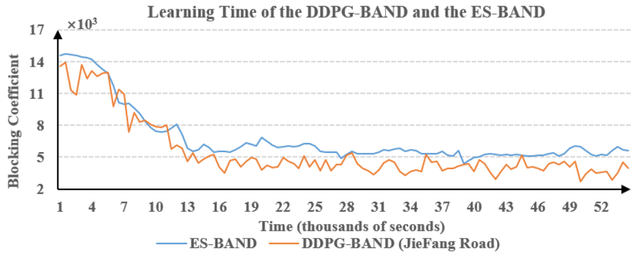

FIGURE 13. The blocking coefficient changes over time during the training process. Since the blocking coefficients at each intersection are trained separately in DDPG-BAND, we use JieFang road as a comparison in this experiment.

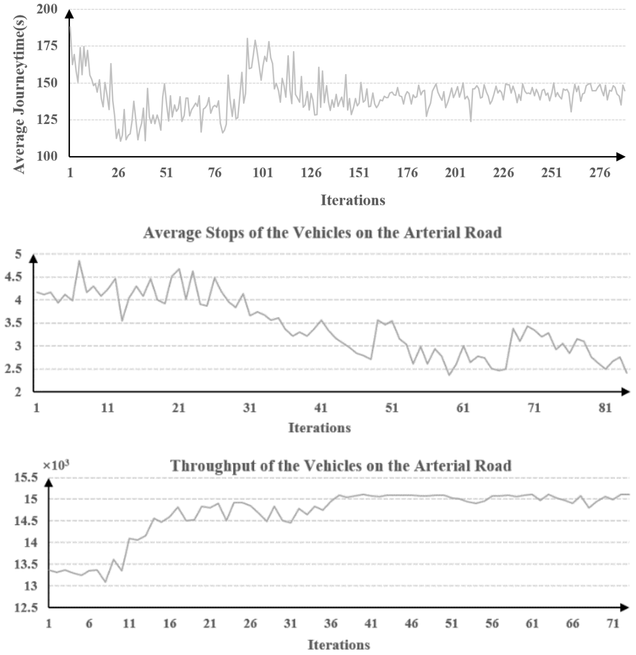

FIGURE 14. Traffic evaluation indexes during the ES-BAND training process.

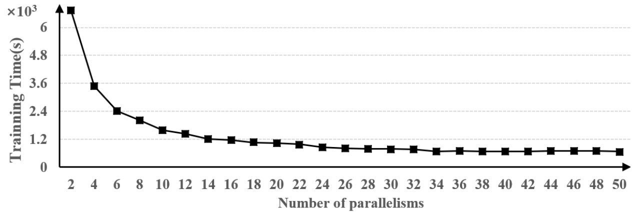

FIGURE 15. Average time spent per iteration with different parallelisms.

sample size λ is represented by Eq. (10):

$$\lambda = 4 + \lfloor 3 \times \log_e N \rfloor \tag{10}$$

The total number of parallel tasks in each iteration is λ multiplied by the number of random seeds. If we have 14 parameters and three random seeds, the number of tasks in each iteration is 33. When 33 cannot exactly divide the parallelism of the process, there will be idle process blocking and the computing resources will not run at full load. If the parallelism of the process rises to 33 or more, the agent will have the fastest learning speed.

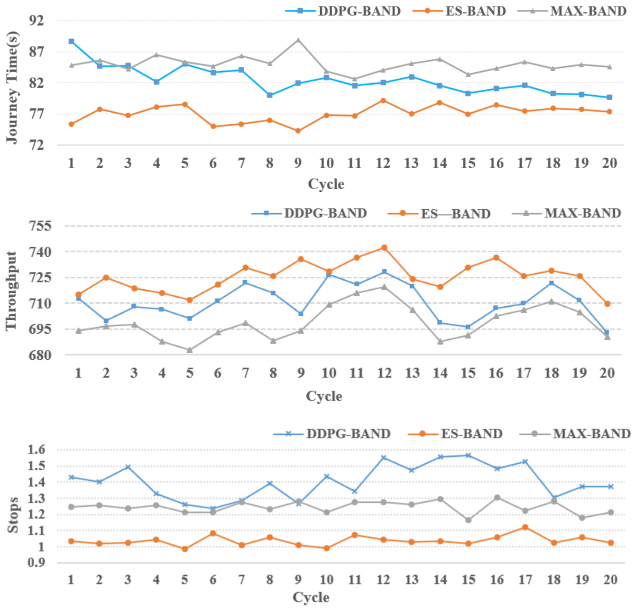

FIGURE 16. The traffic evaluation indexes of MAXBAND, DDPG-BAND, and ES-BAND. In order to avoid biased result, MAXBAND is set with the same cycle length and green split as those of other two AI methods.

F. EVALUATION OF TRAFFIC EVALUATION INDEXES

This group is a test experiment to verify the validity of our approach. In this group of experiments, we compared the results of the traffic evaluation indexes DDPG-BAND and ES-BAND with the benchmark of the traditional MAX-BAND. Considering that the traffic environment of the simulator is different under the different random seeds, we took 10 other random seeds that have never been used in the above training process as the test set. To ensure the test fairness, we performed the following operations on the results of 10 random seeds as the statistical test: 1. The traffic evaluation index under 10 random seeds in each cycle length was calculated, and the maximum and minimum results were removed; 2. The average value of the traffic evaluation index under the remaining 8 random seeds was used to test the final result of traffic evaluation indexes in this cycle. Figure 16 shows the illustration of the traffic evaluation indexes of the traditional MAXBAND, DDPG-BAND, and ES-BAND in each cycle length.

At the beginning of the experiment, to keep a certain number of vehicles on site, we allow the simulator to run for a period of time without calculating the traffic evaluation index; this is called the initial time. In the initial time of the training experiment, the signal timing is random, while in the test set, the traffic signals are controlled by the trained agent. Therefore, whatever the journey time, stops and throughput in the test set experiment are all clearly better than those in the training set experiment, which also proves the effectiveness of artificial intelligence in traffic signal control.

In Fig. 16 we can see that in the 20 periods, the traffic control on ES-BAND has the best performance for average journey time (77.1 s versus 82.47 s in the DDPG-BAND and 85.0 s in the ES-BAND), average number of stop-and-go behaviors (1.04 versus 1.40 in the DDPG-BAND and 1.24 in the ES-BAND), and traffic throughput (725.4 per cycle versus 710.7 per cycle in the DDPG-BAND and 698.3 per cycle in ES-BAND). Therefore, we conclude that, in general, the artificial intelligence method exhibits better performance in traffic evaluation indexes than the traditional mathematical method in our case.

VII. CONCLUSION AND FUTURE WORK

With the development of computer technology, novel traffic control approaches are constantly being proposed to handle urban traffic congestion. This article addresses the deficiencies of the conventional method by proposing two approaches for traffic signal control based on artificial intelligence. DDPG-Band has two stages: a coarse-tune stage and the fine-tune stage. The coarse-tuning stage reduces the blocking coefficient, while the fine-tuning stage optimizes the traffic evaluation index. ES-BAND converts the green light time, offset, and public cycle of traffic light at each intersection into a multidimensional feature vector of a CMA-ES. Optimized training is carried out to obtain the objective of traffic evaluation index. Finally, we successfully applied these algorithms to the green wave traffic control of four consecutive crossovers on Zhongshan West Road in Ningbo, Zhejiang, China and verified their feasibility and effectiveness.

In general, compared with the traditional MAXBAND, the two AI-equipped approaches (DDPG-BAND and ES-BAND) we proposed exhibit better performance no matter in travel time, parking times and throughput for traffic signaling coordination. Furthermore, the experimental results show that ES-BAND has a better coordination effect than DDPG-BAND. This is because the traffic scenario assessed in this article is the same departure probability under different departure rules. DDPG-BAND is more effective in dealing with real-time sequence decision-making problems, that is, it can observe the changes of various variables in traffic scenarios and take corresponding signal action. ES-BAND is good at finding the best solution in a specific environment. It should be noted that the randomness was only introduced to the vehicle departure pattern, and had nothing to do with the amount of traffic flow. Consequently, the introduced randomness was under control and contributed to the enhancement of robustness.

Although the experimental results show that ES-Band exhibits better performance in journey time, stops and throughput, there are still shortcomings in this method. The ES-BAND can provide an accurate fixed timing scheme according to traffic flow, but this means that it is heavily reliant on traffic flow forecasting. ES-BAND is less effective when the traffic flow changes a lot. In our future work, we will concentrate on adaptive traffic signal control in the green wave system. A preliminary idea is that, on the baseline of ES-BAND, we will time the green light within the limited range according to the traffic volume. Therefore, we will start with a neural network where the queue length and road environment in each direction are taken as input data, while the green time will be designed as the output data. We will then search the parameters of neural network by CMA-ES on the simulator. Finally, the complete model, which combines the neural network and ES-BAND, will be obtained for adaptive traffic signal control.

For DDPG-BAND, traffic measurements such as queue length are sensitive to the accuracy of traffic measurements. DDPG-BAND is an adaptive real-time model based on traffic information. Therefore, queue length is needed not only during the training process when it can be obtained from the simulator but also as a parameter of the trained model in the reasoning process. In the reasoning process, real-time traffic measurements obtained from the field are often inaccurate. To avoid this problem, in future work, we will consider using agents to observe fuzzy traffic measurements for training. For example, we can use the queue length level instead of the exact queue length as the observation of the agent. The queue length level changes only when the queue length is increased to a certain extent, thus reducing its sensitivity to the accuracy.

REFERENCES

[1] R. Khatoun and S. Zeadally, "Smart cities: concepts, architectures, research opportunities," *Commun. ACM*, vol. 59, no. 8, pp. 46–57, 2016.

[2] M McKenney, C Frey-Spurlock, "Aging in place: Challenges for smart & resilient communities," in *Proc. 1st ACM SIGSPATIAL Workshop Adv. Resilient Intell. Cities*, Seattle, WA, USA, 2018, pp. 1–2.

[3] J. T. Morgan and J. D. C. Little, "Synchronizing traffic signals for maximal bandwidth," *Oper. Res.*, vol. 12, no. 6, pp. 896–912, Dec. 1964.

[4] J. D. C. Little, "The synchronization of traffic signals by mixed-integer linear programming," *Oper. Res.*, vol. 14, no. 4, pp. 568–594, Aug. 1966.

[5] C. J. Messer, R. H. Whitson, and C. L. Dudek, "A variable-sequence multiphase progression optimization program," *Highway Res. Rec.*, vol. 445, pp. 24–33, Jun. 1973.

[6] N. H. Gartner, J. D. C. Little, and H. Gabbay, "Optimization of traffic signal settings by mixed-integer linear programming: Part I: The network coordination problem," *Transp. Sci.*, vol. 9, no. 4, pp. 344–363, 1975.

[7] R. S. Pillai, A. K. Rathi, and S. L. Cohen, "A restricted branch-and-bound approach for generating maximum bandwidth signal timing plans for traffic networks," *Transp. Res. B, Methodol.*, vol. 32, no. 8, pp. 517–529, Nov. 1998.

[8] N. H. Gartner, S. F. Assman, F. Lasaga, and D. L. Hou, "A multi-band approach to arterial traffic signal optimization," *Transp. Res. B, Methodol.*, vol. 25, no. 1, pp. 55–74, Feb. 1991.

[9] J. Little, M. Kelson, and N. Gartner, "MAXBAND: A program for setting signals on arteries and triangular networks," *Transp. Res. Rec. J. Transp. Res. Board*, vol. 795, pp. 40–46, 1981.

[10] C. P. Pappis and E. H. Mamdani, "A fuzzy logic controller for a trafc junction," *IEEE Trans. Syst., Man, Cybern.*, vol. 7, no. 10, pp. 707–717, Oct. 1977.

[11] C. V. Pham, W. L. Xu, J. Potgieter, F. Alam, F. C. Fang, and W. L. Xu, "A probabilistic fuzzy logic traffic signal control for an isolated intersection," in *Proc. 19th Int. Conf. Mechatronics Mach. Vis. Pract. (M2VIP)*, Nov. 2012, pp. 304–308.

[12] J. Li, M. Dridi, and A. El-Moudni, "A cooperative traffic control for the vehicles in the intersection based on the genetic algorithm," in *Proc. 4th IEEE Int. Colloq. Inf. Sci. Technol. (CiSt)*, Oct. 2016, pp. 627–632.

[13] M. S. Hossain, H. Sinha, and R. Mustafa, "A belief rule based expert system to control traffic signals under uncertainty," in *Proc. Int. Conf. Comput. Inf. Eng. (ICCIE)*, Nov. 2015, pp. 83–86.

[14] P. Mannion and J. E. Duggan Howley, "An experimental review of reinforcement learning algorithms for adaptive traffic signal control," in *Autonomic Road Transport Support Systems*. Berlin, Germany: Springer, 2016, pp. 47–66.

[15] M. Abdoos, N. Mozayani, and A. L. C. Bazzan, "Traffic light control in non-stationary environments based on multi agent Q-learning," in *Proc. 14th Int. IEEE Conf. Intell. Transp. Syst. (ITSC)*, Oct. 2011, pp. 1580–1585.

[16] M. Abdoos, N. Mozayani, and A. L. C. Bazzan, "Hierarchical control of traffic signals using Q-learning with tile coding," *Int. J. Speech Technol.*, vol. 40, no. 2, pp. 201–213, Mar. 2014.

[17] Y. K. Chin, L. K. Lee, N. Bolong, S. S. Yang, and K. T. K. Teo, "Exploring Q-Learning optimization in traffic signal timing plan management," in *Proc. 3rd Int. Conf. Comput. Intell., Commun. Syst. Netw.*, Jul. 2011, pp. 269–274.

[18] S. El-Tantawy, B. Abdulhai, and H. Abdelgawad, "Multiagent reinforcement learning for integrated network of adaptive traffic signal controllers (MARLIN-ATSC): Methodology and large-scale application on Downtown Toronto," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 3, pp. 1140–1150, Sep. 2013.

[19] T. Brys, T. T. Pham, and M. E. Taylor, "Distributed learning and multi-objectivity in traffic light control," *Connection Sci.*, vol. 26, no. 1, pp. 65–83, Jan. 2014.

[20] C. Stamatiadis and N. H. Gartner, "MULTIBAND-96: A program for variable-bandwidth progression optimization of multiarterial traffic networks," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 1554, no. 1, pp. 9–17, Jan. 1996.

[21] R. S. Pillai and A. K. Rathi, "MAXBAND version 3.1: Heuristic and optimal approach for setting the left turn phase sequences in signalized networks," Office Sci. Tech. Inf., Washington, DC, USA, Tech. Rep., 1995, doi: 10.2172/34378.

[22] X. Yang, Y. Cheng, and G.-L. Chang, "A multi-path progression model for synchronization of arterial traffic signals," *Transp. Res. C, Emerg. Technol.*, vol. 53, pp. 93–111, Apr. 2015.

[23] H. S. Tsay and L. T. Lin, "New algorithm for solving the maximum progression bandwidth," in *Proc. 14th Int. IEEE Conf. Intell. Transp. Syst. Conf.*, Washington, DC, USA, 1988.

[24] N. H. Gartner, S. F. Assmann, and F. Lasaga, "MULTIBAND–A variable-bandwidth arterial progression scheme," *Transp. Res. Rec. J. Transp. Res. Board*, vol. 1287, pp. 212–222, 1990.

[25] C. Zhang, Y. Xie, N. H. Gartner, C. Stamatiadis, and T. Arsava, "AM-band: An asymmetrical multi-band model for arterial traffic signal coordination," *Transp. Res. C, Emerg. Technol.*, vol. 58, pp. 515–531, Sep. 2015.

[26] Z. Tian and T. Urbanik, "System partition technique to improve signal coordination and traffic progression," *J. Transp. Eng.*, vol. 133, no. 2, pp. 119–128, Feb. 2007.

[27] Y. Jeong and Y. Kim, "Tram passive signal priority strategy based on the MAXBAND model," *KSCE J. Civil Eng.*, vol. 18, no. 5, pp. 1518–1527, Jun. 2014.

[28] G. Dai, H. Wang, and W. Wang, "A bandwidth approach to arterial signal optimisation with bus priority," *Transportmetrica A: Transp. Sci.*, vol. 11, no. 7, pp. 579–602, Aug. 2015.

[29] J.-Q. Li, "Bandwidth synchronization under progression time uncertainty," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 2, pp. 749–759, Apr. 2014.

[30] N. Hansen, "The CMA evolution strategy: A tutorial," 2016, *arXiv:1604.00772*. [Online]. Available: https://arxiv.org/abs/1604.00772

[31] S. Alam, C. Lokan, G. Aldis, S. Barry, R. Butcher, and H. Abbass, "Systemic identification of airspace collision risk tipping points using an evolutionary multi-objective scenario-based methodology," *Transp. Res. C, Emerg. Technol.*, vol. 35, pp. 57–84, Oct. 2013.

[32] C. E. Cortés, D. Sáez, F. Milla, A. Núñez, and M. Riquelme, "Hybrid predictive control for real-time optimization of public transport systems' operations based on evolutionary multi-objective optimization," *Transp. Res. C, Emerg. Technol.*, vol. 18, no. 5, pp. 757–769, 2010.

[33] X. Kong, G. Shen, F. Xia, and C. Lin, "Urban arterial traffic two-direction green wave intelligent coordination control technique and its application," *Int. J. Control, Autom. Syst.*, vol. 9, no. 1, pp. 60–68, Feb. 2011.

[34] C. Ma and R. He, "Green wave traffic control system optimization based on adaptive genetic-artificial fish swarm algorithm," *Neural Comput. Appl.*, vol. 31, no. 6, pp. 2073–2083, 2015.

[35] Y. Gao, D. Jiang, and Y. Xu, "Optimize taxi driving strategies based on reinforcement learning," *Int. J. Geographical Inf. Sci.*, vol. 32, no. 8, pp. 1677–1696, Aug. 2018.

[36] Y. Hu, Q. Da, A. Zeng, Y. Yu, and Y. Xu, "Reinforcement learning to rank in e-commerce search engine: Formalization, analysis, and application," in *Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2018, pp. 368–377.

[37] R. S. Sutton and A. G. Barto, "Reinforcement learning: An introduction," *IEEE Trans. Neural Netw.*, vol. 9, no. 5, p. 1054, Sep. 1998.

[38] W. Genders and S. Razavi, "Using a deep reinforcement learning agent for traffic signal control," 2016, *arXiv:1611.01142*. [Online]. Available: https://arxiv.org/abs/1611.01142

[39] I. Jang, D. Kim, D. Lee, and Y. Son, "An agent-based simulation modeling with deep reinforcement learning for smart traffic signal control," in *Proc. Int. Conf. Inf. Commun. Technol. Converg. (ICTC)*, Oct. 2018, pp. 1028–1030.

[40] B. Bakker, S. Whiteson, L. Kester, and F. C. A. Groen, "Traffic light control by multiagent reinforcement learning systems," in *Interactive Collaborative Information Systems*. Berlin, Germany: Springer, 2010, pp. 475–510.

[41] M. L. Littman, "Reinforcement learning improves behaviour from evaluative feedback," *Nature*, vol. 521, no. 7553, pp. 445–451, May 2015.

[42] B. Abdulhai, R. Pringle, and G. J. Karakoulas, "Reinforcement learning for true adaptive traffic signal control," *J. Transp. Eng.*, vol. 129, no. 3, pp. 278–285, May 2003.

[43] Z. Ning, R. Y. K. Kwok, K. Zhang, X. Wang, M. S. Obaidat, L. Guo, X. Hu, B. Hu, Y. Guo, and B. Sadoun, "Joint computing and caching in 5G-envisioned Internet of vehicles: A deep reinforcement learning-based traffic control system," *IEEE Trans. Intell. Transp. Syst.*, early access, Feb. 5, 2020, doi: 10.1109/TITS.2020.2970276.

[44] P. Zhou, T. Braud, A. Alhilal, P. Hui, and J. Kangasharju, "ERL: Edge based reinforcement learning for optimized urban traffic light control," in *Proc. IEEE Int. Conf. Pervas. Comput. Commun. Workshops (PerCom Workshops)*, Mar. 2019, pp. 849–854.

[45] H. Joo, S. H. Ahmed, and Y. Lim, "Traffic signal control for smart cities using reinforcement learning," *Comput. Commun.*, vol. 154, pp. 324–330, Mar. 2020.

[46] K. L. Tan, "Deep reinforcement learning for adaptive traffic signal control," in *Proc. ASME Dyn. Syst. Control Conf.*, Park City, UT, USA, Oct. 2019.

[47] X. Wang, L. Ke, Z. Qiao, and X. Chai, "Large-scale traffic signal control using a novel multiagent reinforcement learning," *IEEE Trans. Cybern.*, Sep. 3, 2020, doi: 10.1109/TCYB.2020.3015811.

[48] X. Liang, X. Du, G. Wang, and Z. Han, "A deep reinforcement learning network for traffic light cycle control," *IEEE Trans. Veh. Technol.*, vol. 68, no. 2, pp. 1243–1253, Feb. 2019.

[49] A. Cabrejas-Egea, S. Howell, M. Knutins, and C. Connaughton, "Assessment of Reward Functions for Reinforcement Learning Traffic Signal Control under Real-World Limitations," 2020, *arXiv:2008.11634*. [Online]. Available: https://arxiv.org/abs/2008.11634

[50] R. Zhang, A. Ishikawa, W. Wang, B. Striner, and O. K. Tonguz, "Using reinforcement learning with partial vehicle detection for intelligent traffic signal control," *IEEE Trans. Intell. Transp. Syst.*, early access, Mar. 30, 2020, doi: 10.1109/TITS.2019.2958859.

[51] Y. Xiong, G. Zheng, K. Xu, and Z. Li, "Learning traffic signal control from demonstrations," in *Proc. 28th ACM Int. Conf. Inf. Knowl. Manage.*, Nov. 2019, pp. 2289–2292.

[52] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, Feb. 2015.

[53] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," 2015, *arXiv:1509.02971*. [Online]. Available: https://arxiv.org/abs/1509.02971

[54] H. Xie, E. Tanin, S. Karunasekera, L. Kulik, R. Zhang, J. Qi, and K. Ramamohanarao, "Studying transportation problems with the SMARTS simulator (demo paper)," in *Proc. 26th ACM SIGSPATIAL Int. Conf. Adv. Geographic Inf. Syst.*, Seattle, WA, USA, 2018, pp. 580–583.

[55] P. A. Lopez, M. Behrisch, L. Bieker-Walz, J. Erdmann, Y.-P. Flötteröd, R. Hilbrich, and L. Lücken, "Microscopic traffic simulation using SUMO," in *Proc. 21st Int. Conf. Intell. Transp. Syst. (ITSC)*, Maui, HI, USA, Nov. 2018, pp. 4–7.

[56] N. Omar, J. Prasetijo, B. D. Daniel, M. A. E. Abdullah, and I. Ismail, "Study of car acceleration and deceleration characteristics at dangerous route FT050," in *Proc. IOP Conf., Earth Environ. Sci.*, vol. 140, Apr. 2018, Art. no. 012078.

[57] A. Mehar, S. Chandra, and S. Velmurugan, "Speed and acceleration characteristics of different types of vehicles on multi-lane highways," *Eur. Transp.*, vol. 55, no. 1, pp. 1–12, 2013.

[58] H. Summala, "Brake reaction times and driver behavior analysis," *Transp. Hum. Factors*, vol. 2, no. 3, pp. 217–226, Sep. 2000.

[59] P. S. Bokare and A. K. Maurya, "Acceleration-deceleration behaviour of various vehicle types," *Transp. Res. Procedia*, vol. 25, pp. 4737–4753, 2017.