Hindawi
Journal of Advanced Transportation
Volume 2019, Article ID 1019078, 12 pages
https://doi.org/10.1155/2019/1019078

Research Article

Hierarchical Sarsa Learning Based Route Guidance Algorithm

Feng Wen,[1] **Xingqiao Wang**,[1] **and Xiaowei Xu**[1,2]

[1]*School of Information Science and Engineering, Shenyang Ligong University, Shenyang, Liaoning, China*
[2]*Department of Information Science, University of Arkansas Little Rock, Little Rock, Arkansas, USA*

Correspondence should be addressed to Feng Wen; wenfeng@sylu.edu.cn

Received 24 December 2018; Revised 27 February 2019; Accepted 14 March 2019; Published 27 June 2019

Academic Editor: Alain Lambert

In modern society, route guidance problems can be found everywhere. Reinforcement learning models can be normally used to solve such kind of problems; particularly, Sarsa Learning is suitable for tackling with dynamic route guidance problem. But how to solve the large state space of digital road network is a challenge for Sarsa Learning, which is very common due to the large scale of modern road network. In this study, the hierarchical Sarsa learning based route guidance algorithm (HSLRG) is proposed to guide vehicles in the large scale road network, in which, by decomposing the route guidance task, the state space of route guidance system can be reduced. In this method, Multilevel Network method is introduced, and Differential Evolution based clustering method is adopted to optimize the multilevel road network structure. The proposed algorithm was simulated with several different scale road networks; the experiment results show that, in the large scale road networks, the proposed method can greatly enhance the efficiency of the dynamic route guidance system.

1. Introduction

In the recent decades, more and more people own their private vehicles, and the traffic pressure in the city increased rapidly. Citizens' life quality is always undermined by daily delay which is one of the consequences of traffic congestion. The congestion can also cause the aggravation of pollution and the increasing of travelling cost. The dynamic route guidance method, which can not only provide travel routes but also relieve the traffic congestion, attracted many scholars' attention [1–3].

Dynamic route guidance system (DRGS) is an important part of Intelligent Transportation System (ITS), in which centrally determined route guidance system (CDRGS) [4] is economically effective and efficient for drivers and can avoid Braess's paradox [5]. CDRGS guides all the vehicles for all the possible origin destination (OD) pairs with the real-time information and considers guidance in terms of the whole traffic system. However, traditional route guidance methods, like Dijkstra Algorithm [6] and A∗ Algorithm[7], are not suitable in the dynamic traffic environment [8], because these shortest path algorithms may cause traffic concentration and overreaction phenomenon when they are adopted to guide plenty of vehicles. Multiple paths routing algorithm [9] could relief the traffic jam by distributing traffic into different paths and does not depend too much on the real-time data, but when it needs to compute new solutions, the response time may be lengthened. Reinforcement learning strategy has been widely used in the dynamic environment [10–13], because it can reduce the computational time and make full use of real-time information. With these characters, reinforcement learning strategy has been used in the dynamic route guidance system. Shanqing et al. [14] applied Sarsa learning to guide vehicles in the dynamic environments by considering minimizing the route computational time. In our earlier study [15], Sarsa learning is adopted to guide vehicles in CDRGS and the Boltzmann distribution is selected as the action selection method. The results show that, compared with traditional methods, the proposed Sarsa learning based route guidance algorithm (SLRGA) and Sarsa learning with Boltzmann distribution algorithm (SLWBD) can strongly reduce the travelling time and relieve traffic congestion.

However, the scale of real-world road networks is usually large, and then the scale of state set of reinforcement learning based route guidance system responding to these road networks is huge. Thus it is really difficult for reinforcement

learning based route guidance system to be convergent in the larger scale traffic environment. So, how to solve the route guidance problem in the large scale road network with reinforcement learning method is a challenge. Hierarchical reinforcement learning (HRL) can improve in both time and searching space for learning and execution of the whole task by recursively decomposing larger and complex tasks into sequentially executed smaller and simpler subtasks [13]. The decomposition strategy is a key point in the hierarchical context [16], and when HRL is used in solving the route guidance problem in the large scale road networks, avoiding congestion phenomenon and reducing vehicles' traveling time can be achieved by an effective decomposition of the route guidance.

Heng Ding et al. [17] proposed a macroscopic fundamental diagram (MFD) based traffic guidance perimeter control coupled (TGPCC) method to improve the performance of macroscopic traffic networks. They establish a programming function according to the network equilibrium rule of traffic flow amongst multiple MFD subregions, which reduce the congestion phenomenon by effectively assigning the traffic flow amongst different subregions. So, partitioning the original network and assigning traffic flows in subnetworks are effectively considered as the objective of the decomposition strategy when HRL is adopted for solving route guidance problems.

Multilevel approach has been successfully employed in a variety of problems [18] and Multilevel Network method [19] is considered to be introduced to segment the original network into several subnetworks and generate higher level network. S. Jung et al. [20] indicated that the optimal route on higher level network between two nodes is equivalent to that on original road network. Thus, Multilevel Network method can be utilized to perform the route guidance task in the large scale road network, in which route guidance on the higher level network can be seen as the decomposition of the route guidance task, and as a result, this method would not affect the preciseness of route guidance.

Therefore, Multilevel Network structure based HRL is adopted in this study, and considering the on-line learning characteristic of Sarsa learning method and its effective performance in solving route guidance problems[15], the hierarchical Sarsa learning based route guidance algorithm (HSLRG) is proposed to guide vehicles with proper routes in the large scale road network. The route guidance task can be divided into several smaller route guidance tasks, and then these smaller route guidance tasks perform on the corresponding subnetworks. To generate the Multilevel Network structure, traditional clustering methods like K-means [21] and K-modes [22] have been considered. However comparing with conventional clustering methods, evolution based clustering method can avoid tripping into local optimal problem [19]. In addition, evolutionary algorithm can always deal with multiobjective problems effectively [23–26]. In this study, Differential Evolution [27, 28] based clustering method, which can be adopted in complex environment [29], is introduced, and multiobjective functions are designed to optimize the Multilevel Network structure.

The contribution of this work is shown as follows: Firstly, we proposed a novel Multilevel Network structure based dynamic route guidance method. By reducing the state action space with Multilevel Network structure, the route guidance method can greatly reduce the congestion phenomenon in the road network and improve the efficiency of the whole transportation system notably. Secondly, we provide a Differential Evolution based clustering method to construct the Multilevel Network with multiobjectives. These objectives consider optimizing the structure from both higher level network and subnetwork aspects and optimize the structure greatly.

This paper includes seven sections. Section 2 introduces the Multilevel Network based route guidance model (MNRGM). Section 3 introduces the Differential Evolution based clustering method. Section 4 proposes HSLRG and describes the main procedure and details of it. Section 5 introduces the experimental conditions and discusses and analyzes the results. The last parts of this paper are the conclusion and acknowledgement sections.

2. Multilevel Network Based Route Guidance Model

In this section, MNRGM is introduced. HRL can reduce the searching space, and in this study, it is used to decompose the vehicle guidance from the original network into subnetworks. Sarsa learning, which fits for solving dynamic environment problems [30, 31], is adopted to guide vehicles in the Multilevel Network. The purpose of this model can be seen as follows:

(i) Reduce the average travelling time of vehicles in the large scale road network.

(ii) Reduce the probability of congestion in the large scale road network.

(iii) Reduce the searching space of reinforcement learning in the large scale road.

And we assumed that the real-time travelling information in the Multilevel Network can be collected.

2.1. Multilevel Network Model. Multilevel Network is constructed by dividing the original network into several subnetworks. The example of two-level network can be seen as Figure 1. The boundary nodes of subnetworks and the optimal routes between them are nodes and links on higher level network.

In this model, the topographical road map is seen as the directed network $G(V, E)$, where V denotes the set of nodes of road network and E denotes the set of links of road network; i.e., s_{ij} corresponds to the link from node i to node j. The cost of it in this model is measured by the traveling time. If $G(V, E)$ can be divided into m subnetworks like $G_1(V_1, E_1), G_2(V_2, E_2), \ldots, G_m(V_m, E_m)$ then

$$V = V_1 \cup V_2 \cup \cdots \cup V_m,$$
$$E = E_1 \cup E_2 \cup \cdots \cup E_m \quad (1)$$

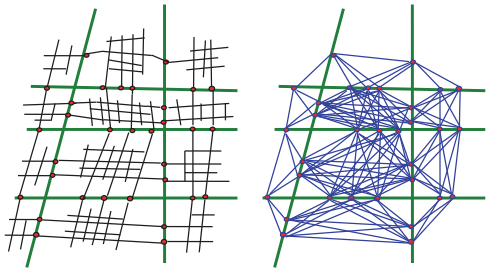


Figure 1: An example of Multilevel Network.

In the subnetwork, the nodes can be divided into two categories: interior nodes and boundary nodes. A node is a boundary node if it belongs to more than one subnetwork, and vice versa.

The Multilevel Network model is shown as follows.

Indices. $i, j, r \in \{1, 2, \ldots, n\}$, index of node.

Parameters

n: the number of nodes.

o: origin node.

d: destination node.

$R(o, d)$: a route from o to d.

s_{ij}^k: link from node i to node j in level k.

k: index of the level in Multilevel Network, $k \in \{1, 2, \ldots, K_{max}\}$.

K_{max}: the maximum level of Multilevel Network.

n_k: the number of nodes in level k of Multilevel Network.

c_{ij}^k: cost of link s_{ij} in level k of Multilevel Network.

$F(r)$: set of nodes connected from node r.

$T(r)$: set of nodes connected to node r.

Decision Variables

$$x_{ij}^k = \begin{cases} 1, & \text{if and only if link } s_{ij} \text{ is included in } R(o,d) \text{ in level } k \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

The optimal path on Multilevel Network can be calculated as follows:

$$\min \sum_{k=1}^{K_{max}} \sum_{i=1}^{n_k} \sum_{j=1}^{n_k} c_{ij}^k x_{ij}^k \quad (3)$$

$$\text{s.t.} \sum_{j \in F(r)} x_{rj}^k - \sum_{i \in T(r)} x_{ir}^k = \begin{cases} 1 & (r = o) \\ 0 & (r \in V \setminus \{o, d\}) \\ -1 & (r = d) \end{cases} \quad (4)$$

$$x_{ij}^k \in \{0, 1\}, \quad \forall i, j, k \quad (5)$$

where constraints (4) and (5) can ensure the flow conservation rule to be observed for $V \setminus \{o, d\}$.

The K_{max} is set as 2 in the simulations of this study.

We use $G_{high}(V', E')$ to represent the higher level network, where V' and E' are the set of nodes and links of higher level network, respectively.

The set of boundary nodes between any subnetworks $G_i(V_i, E_i)$ and $G_j(V_j, E_j)$ is $V_i \cap V_j$, where $i \neq j$. We use $B(G_i)$ to represent the set of boundary nodes of subnetwork $G_i(V_i, E_i)$. Then,

$$B(G_i) = \bigcup_{j=1}^{m} (V_i \cap V_j), \quad \text{where } i = 1, 2, \ldots, m, \ j \neq i. \quad (6)$$

Let B^T represent the set of the boundary nodes:

$$B^T = \bigcup_{i=1}^{m} B(G_i) \quad \text{where } V' = B^T. \quad (7)$$

Links of the higher level network are calculated and generated based on B^T. In $G_i(V_i, E_i)$, we use $l(u, v)$ to represent the optimal route between any node pair u and v in $B(G_i)$; the cost function $f_c(u, v)$ of $l(.)$ is shown as follows:

$$f_c(u, v) = \begin{cases} l(u, v) & \text{if there is a route from } u \text{ to } v \text{ on } G_i(V_i, E_i) \text{ without any other boundary node on the route;} \\ \emptyset & \text{otherwise.} \end{cases} \quad (8)$$

For subnetwork $G_i(V_i, E_i)$, let

$$L(G_i) = \{(\boldsymbol{u}, \boldsymbol{v}) \mid (\boldsymbol{u}, \boldsymbol{v}) \in \{(B(G_i) \times B(G_i))\}\} \quad (9)$$

Let L^T represent the set of links of the higher level network:

$$L^T = \bigcup_{i=1}^{m} L(G_i) \quad \text{where } E' = L^T \quad (10)$$

In order to guide vehicles in this structure, once the OD pairs are determined, the higher level network is extended, the extension of higher level network can be denoted as $G'_{high}(B^{T'}, L^{T'})$, where $B^{T'}$ is the extension of B^T, which can be shown as $B^{T'} = B^T \cup O \cup D$, and $L^{T'}$ is the extension of L^T, which is shown as $L^{T'} = L^T \cup L(O) \cup L(D)$, $L(O)$

denotes the set of routes from original node to boundary nodes in the corresponding subnetwork, and $L(D)$ denotes the set of routes from boundary nodes to destination node in the corresponding subnetworks, which can be shown as

$$L(O) = \{(o,u) \mid (o,u) \in \{(O \times B(G_i)\}\} \tag{11}$$

$$L(D) = \{(u,d) \mid (u,d) \in \{(B(G_j) \times D\}\} \tag{12}$$

where O is the set of original nodes, D is set of destination nodes, and G_i and G_j are the corresponding subnetworks of O and D.

2.2. Multilevel Network Based Hierarchical Reinforcement Learning

2.2.1. Hierarchical Sarsa Learning. Hierarchical reinforcement learning (HRL)[32] decomposes a reinforcement learning task into a hierarchy of subtasks so that lower-level child tasks can be invoked by higher-level parent tasks to reduce computing time and searching space.

In this study, the route guidance tasks are decomposed according to the structure of the Multilevel Network. As shown in Figure 2, the guidance in the higher level network (the selected series of links in the higher level network) determines the subtasks in the subnetworks. It guides vehicles from a node in the subnetwork to a boundary node or a destination node in this subnetwork. For example, as shown in Figure 3, the vehicle guidance on the original network is decomposed into guidance on three subnetworks, which can be seen as follows:

(i) Vehicle departs from original node O and arrives at boundary node B_i in subnetwork G_1;

(ii) Vehicle departs from boundary node B_i and arrives at boundary node B_j in subnetwork G_2;

(iii) Vehicle departs from boundary node B_j and arrives at destination node D in subnetwork G_3.

In the hierarchical Sarsa learning model, the agent is the CDRGS in each road network (both subnetworks and higher level network), and the purpose of the CDRGS is to guide all the vehicles in the traffic road network and to pursue the optimal travelling time. For each agent, the state is continuous, which is the positions and destinations of all the vehicles in the corresponding subnetwork (or higher level network); the description of the continuous state space of any graph G_i can be shown as follows:

$$State_c(G_i) = \left(\left(p(vel_1), d(vel_1)\right), \ldots, \left(p\left(vel_j\right), d\left(vel_j\right)\right), \ldots\right) \tag{13}$$

where G_i is the ith subnetwork, $vel_j \in VEL(G_i)$ are the vehicles in G_i, $p(vel_j)$ is the position of vehicle vel_j, and $d(vel_j)$ is the destination of vehicle $d(vel_j)$.

In order to reduce the state space, the discrete states which are the nodes and destinations of each vehicle are adopted. In the original network, the state space is $State_d(G)$; with the Multilevel Network structure, the state space is reduced, each subnetwork has the state space $State_d(G_i)$, the state space of higher level network is $State_d(G_{high})$, and the function can be seen as follows:

$$State_d(G_i) = \left(\left(v(vel_1), d(vel_1)\right), \ldots, \left(v\left(vel_j\right), d\left(vel_j\right)\right), \ldots\right) \tag{14}$$

where G_i is the ith subnetwork, $vel_j \in VEL(G_i)$ are the vehicles in subnetwork G_i, $v \in V_i$, V_i is the set of nodes in subnetwork G_i, $v(vel_j)$ is the nearest node in front of vehicle vel_j, and $d(vel_j)$ is the destination node of vehicle $d(vel_j)$.

The action of each agent is an array which is composed of selections of next guided link of each vehicle, which is shown as follows:

$$Action(G_i) = \left(e(vel_1), \ldots, e\left(vel_j\right), \ldots\right) \tag{15}$$

where $e(.) \in E_i$ is the guided next link of vehicle, and E_i is the set of links in subnetwork G_i.

According to the $Action(G_i)$, as shown in Figure 4, in each network (both higher level network and subnetwork), vehicles would receive their guidance information. And the passing time which is the time spent by each vehicle in the corresponding link composes the penalty; the penalty can be seen as follows:

$$P(G_i) = \left(t(vel_1), t(vel_2), \ldots, t\left(vel_j\right), \ldots\right) \tag{16}$$

where $t(vel_j)$ is passing time of vehicle vel_j for the link $e(vel_j)$.

Q-value matrix is used to guide vehicles in each subnetwork and higher level network, in which each Q-value represents the estimate optimal traveling time from the corresponding link to the destination. The proposed vehicle guidance method on both level networks is based on Sarsa learning. The equation of updating Q-values in the matrix with Sarsa learning method is shown as follows:

$$Q_d(i,j) \longleftarrow Q_d(i,j) + \alpha * \left(t_{ij} + \gamma * Q_d(j,k) - Q_d(i,j)\right) \tag{17}$$

where $Q_d(i,j)$ is the estimated optimal traveling time to destination d for each vehicle which selects moving to node j in node i; t_{ij} is the travelling time of the latest passing time of link s_{ij}; k is the node belonging to $F(j)$ (the set of nodes connected from node j), through which vehicles travel to destination d after they passed link s_{ij}; α is the learning rate. γ is the discount rate.

Boltzmann distribution [33] is adopted as the probability distribution of action selection in this study which can balance the exploration and exploitation of action selection according to the Q-values. The probability model of action selection is shown as follows:

$$p_d(i,j) \longleftarrow \frac{e^{-(1/\tau)(Q_d(i,j)/EQ_d(i))}}{\sum_{j \in A(i)} e^{-(1/\tau)(Q_d(i,j)/EQ_d(i))}} \tag{18}$$

where $EQ(i)$ is the average Q-value from node i to destination d; τ is *temperature*.

$$\tau = \frac{\tau_{max}}{1 + e^{-\alpha(NV-\beta)}} \tag{19}$$

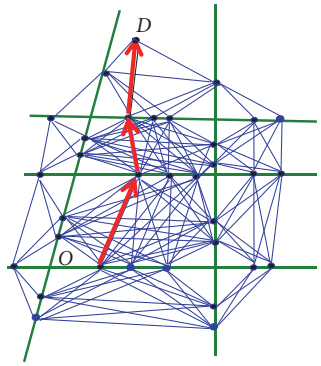


Figure 2: An example of vehicle guidance in the higher level network.

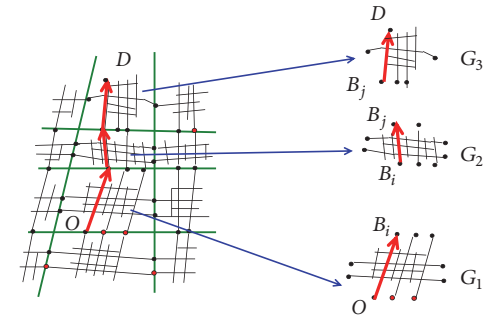


Figure 3: An example of decomposition of route guidance.

where $\tau_{max}, \alpha, \beta$ are constants; NV is the total number of vehicles in the road network.

2.2.2. Optimizing Multilevel Network Structure. In this study, in order to accelerate the convergence of reinforcement learning in the Multilevel Network, the structure of the Multilevel Network should be considered. Both state action space of subnetworks and higher level network can be optimized with clustering method. Two objective functions have been considered, which are described as follows:

$$\sum \left|B_i^T\right| * S\left(G_i\right) \tag{20}$$

$$S\left(G_{high}\right) \tag{21}$$

where $S(.)$ is the searching space of the road network, and it can be calculated as follows:

$$S\left(G\right) = \prod_{i=1}^{|V|} E\left(v_i\right) \tag{22}$$

where $E(v)$ is the number of links departing from node v if the set is not null; otherwise it is 1.

3. Differential Evolution Based Clustering Method

Ding et al. [17] divided the heterogeneous networks into homogeneous subregions, which have small variances in link densities, such that each subregion has a well-defined MFD shape. In the proposed method, multiple homogeneous similar scale subnetworks and a virtual higher level network which can effectively assign traffic flows among them are required. In this section, a Differential Evolution based clustering method is used to generate the previous Multilevel Network structure offline.

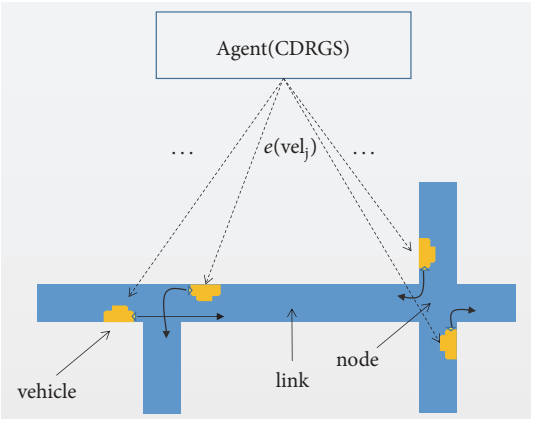


Figure 4: Demonstration of vehicle guidance in the network.

3.1. DE Based Clustering Method. DE [27, 28] is a well-known direction based evolution method which can search the optimal solution effectively in large scale searching space. In order to construct the proper Multilevel Network structure, various individuals should be maintained in the population, and an effective evolution direction is necessary. Thus DE is selected as the clustering method.

In the proposed method, decoding operator is clustering the road network, and after decoding, each gene in the chromosome becomes a subnetwork. On the other word, subnetwork $G_i(V_i, E_i)$ is cluster i of the clustering result of the corresponding chromosome.

In order to accelerate the convergence of reinforcement learning in the Multilevel Network, two factors are considered when the Multilevel Networks are constructed. The first one is the convergence efficiency of reinforcement learning on each subnetwork. The second one is the convergence efficiency of reinforcement learning on the higher level network. Therefore, there are two objective functions, minimizing the state action space of all subnetworks in (23) and minimizing the state action space of the higher level network in (24).

$$\sum \left|B_i^T\right| * S\left(G_i\right) \tag{23}$$

$$S\left(G_{high}\right) \tag{24}$$

In order to achieve these two objective functions simultaneously, a fitness function is used, which is shown as follows:

$$Fitness = \log \sum \left|B_i^T\right| * S\left(G_i\right) + \log S\left(G_{high}\right) \tag{25}$$

3.2. Genetic Representation. When the Multilevel Network structure is constructed by the DE based clustering method,

the number of clusters has strong influence on the number of nodes and links of the higher level network [34], which will affect the two objective functions. So, an appropriate number of clusters should be found to optimize the structure of the Multilevel Network.

In this study, in order to get the proper number of clusters, two vectors, coordinate value vector and available vector, are defined in the chromosome. Each element in the coordinate value vector is corresponding to the element in the same position of the available vector. The maximum length of these vectors is M, the coordinate values vectors present cluster centroids, and each number of the available vector represents the validity of the corresponding centroid; if the number is bigger than the threshold *valid*, the corresponding centroid is valid, and visa versa.

The decoding procedure is the clustering procedure, in which the Multilevel Network structure is generated with each valid gene.

3.3. Differential Evolution. The DE operator of any individual x_i can be seen as follows:

$$l_i = r_1 + F\left(r_2 - r_3\right), \quad r_1 \neq r_2 \neq r_3 \neq x_i \tag{26}$$

where r_1, r_2, and r_3 are three different individuals which are randomly selected from the population, l_i is the mutants of x_i, (r_2-r_3) forms a vector, and F which is a positive real number controls the length of the vector.

The overall procedure of DE based clustering method can be seen in Algorithm 1.

4. Hierarchical Sarsa Learning Based Route Guidance Algorithm

4.1. Overall Procedure. After generating the optimized Multilevel Network structure, the proposed hierarchical Sarsa learning based route guidance algorithm (HSLRG) can be divided into 3 stages:

(i) Initializing stage: initialize Q-values of all the boundary nodes and destination nodes in the Multilevel Network.

(ii) Route guidance stage: guide vehicles in the higher level network and subnetworks.

(iii) Updating stage: update Q-values of all the boundary nodes and destination nodes in the Multilevel Network.

Before each updating stage, the CDRGS collects travelling information from the environment. During the period, the CDRGS guides vehicles with the Q-values updated in last updating stage. The overall procedure of the proposed HSLRG is shown as Algorithm 2.

4.2. Initializing Q-Values. Q-value based Dynamic Programming is adopted to initialize the Q-values of Sarsa of the Multilevel Network, and Q-values are iteratively calculated by the following equation.

$$Q_d^{(n)}(i, j) = t_{ij}^c + \min_{k \in F(j)} Q_d^{(n-1)}(j, k) \quad i \in I - d - B(d), \ j \in F(d) \tag{27}$$

where $i, j \in I$ is set of nodes; $d \in D$ is set of destinations; s_{ij} is link departure from node i to node j; t_{ij}^c is the history traveling time of link s_{ij}; $F(i)$ is set of nodes depart from node i.

In this study, the procedure of initialization can be seen as Algorithm 3.

4.3. Route Guidance Procedure. In the HSLRG, the guidance is based on the Sarsa learning in the Multilevel Network. The guidance in the higher level network determines the actual destinations of vehicles in each subnetwork. The route guidance procedure for each vehicle of CDRGS can be divided into 3 steps, which can be seen as follows.

Step 1. Guide vehicle in the higher level network with Algorithm 4 and get the selected link (the subtask on the subnetwork).

Step 2. According to the result of Step 1, guide vehicle in the subnetwork with Algorithm 4 until the vehicle reaches the boundary node or destination.

Step 3. If the vehicle does not reach destination, turn to Step 1.

4.4. Updating Procedure. During the updating stage, the following steps should be performed:

(i) Update the Q-values of $d \in B^{T'}$ in each subnetwork G_i.

(ii) Update the Q-values of $d \in D$ in the higher level network G'_{high}.

The procedure of updating is presented as Algorithm 5.

The updates of Q-value for each subnetwork/high level network are independent of each other, so the updating of the proposed method is designed computing parallel, and the time complexity of updating stage is $O(|D_G| * |L_G|)$, where, $|D_G|$ and $|L_G|$ are the number of elements in destination set and link set in the road network G, respectively.

5. Simulation

In this study, the SUMO [35] simulator is used to implement the experiments with three different digital road networks as shown in Table 1. All the algorithms were coded in Java and a PC with 8-core Xeon E5-2640 v3 2.60GHZ processor and 128GB of RAM running Linux (centos 6.6) was used for the all experiments. Our experiments are conducted using real networks, representing various roads of Japan (Experiment 1 and Experiment 2) and US (Experiment 3). The Japan digital road maps are taken from Japan Digital Road Map Association (JDRMA). The US digital networks

Table 1: Data of experiments.

Item	Experiment 1	Experiment 2	Experiment 3
Number of nodes	1500	1800	3500
Number of links	4620	5488	11310
Number of OD	33	33	100
Number of OD pairs	1089	1089	10000
Vehicle departure rate of each origin node	7 seconds per vehicle	7 seconds per vehicle	8 seconds per vehicle

input: road network data, DE parameters
output: optimal solutions E(P)
begin
current generation $t \longleftarrow 0$;
initialize Population $P(t)$;
generate Multilevel Network according to each chromosome;
evaluate each Multilevel Network;
while not termination condition **do**
  for individual x_i^t **do**
    if random(0,1)< P_C **then**
      selected different individuals r_1^t, r_2^t, r_3^t from $P(t)$ randomly;
      $l_i^t = r_1^t + F(r_2^t - r_3^t)$
      generate a Multilevel Network according to the chromosome;
      evaluate the Multilevel Network;
    end if
  end for
  $t \longleftarrow t + 1$;
  for individual x_i^t **do**
    if $Fitness(l_i^{t-1}) < Fitness(x_i^{t-1})$, $x_i^t = l_i^{t-1}$;
    otherwise $x_i^t = x_i^{t-1}$.
  end for
end while
end

Algorithm 1: Procedure of DE based clustering.

begin
//Initializing stage
Initialization routine
while not termination condition **do**
  while at updating interval **do**
    //Route Guidance stage
    Route Guidance routine
  end while
  //Updating stage
  Updating routine;
end while
end

Algorithm 2: Overall procedure of Hierarchical Sarsa Learning based route guidance algorithm.

is provided by the Topologically Integrated Geographic Encoding and Referencing (TIGER)/line collection, available at http://www.diag.uniroma1.it/challenge9/data/tiger/. In the simulation, a time step means a second, and the length of simulation of experiments is set as 15000 time steps.

Table 2: Results of DE based clustering method.

Item	Experiment 1	Experiment 2	Experiment 3
Number of clusters	12	11	21
Fitness	228	229	607

5.1. Multilevel Network. DE based clustering method is used to generate Multilevel Network of each experiment, the evolution process can be seen as Figure 5, the x-axis is the generation, and the y-axis is the average fitness of individuals in the population. The results of the DE can be seen as Table 2.

It can be seen that the DE based clustering method can reduce the fitness during the process of evolution effectively and Multilevel Network structure which is used in the proposed algorithm has been optimized greatly.

5.2. Comparing Method. In the experiments, the Dijkstra algorithm (DA) and Sarsa learning based route guidance on the original road network method are adopted to compare with the proposed method.

(1)Dijkstra algorithm (DA): DA is adopted to represent the static shortest route method, and it calculates the routes

begin
//Initializing Q-value of $B^{T'}$ in each subnetwork
for each $d \in B^{T'}$ **do**
  Initialize Q_d According to Eq. (27) in the corresponding subnetwork
end for
//Initializing Q-value of D in the higher level network
for each $d \in D$ **do**
  Initialize Q_d According to Eq. (27) in the G'_{high}
end for
end

Algorithm 3: Procedure of Initialization.

input: Vehicle v, Destination d
output: Next link s_{jk}
begin
Get the link s_{ij} link of vehicle v
//Calculating the probabilities of next links according to Eq. (18).
$$p_d(j,k) \longleftarrow \frac{e^{-(1/\tau)(Q_d(j,k)/EQ_d(j))}}{\sum_{k \in A(j)} e^{-(1/\tau)(Q_d(j,k)/EQ_d(j))}}$$
//Selecting the next link.
Choose s_{jk} by $p_d(j,k)$.
end

Algorithm 4: Procedure of Route Guidance.

every 60 time steps based on real-time traffic information which is supposed to be collected in this study.

(2)Sarsa learning based route guidance on the original road network method: In order to evaluate the efficiency of Multilevel Network based route guidance method, Sarsa learning with Boltzmann distribution algorithm (SLWBD), which only considers the route guidance on the original road network, is adopted as comparing method in the simulations. The Boltzmann distribution is selected as the action selection method. The Q-values are updated with (17) every 60 time steps.

5.3. Evaluation. Two kinds of criteria are adopted to evaluate the performance of route guidance algorithm.

(1) The number of vehicles in the traffic system $NumV$;

(2) The average traveling time of vehicles arriving destinations in the a period of time, which is calculated as follows:

$$aveT(t) = \sum_{i=1}^{N(t)} \frac{T(v_i)}{N(t)} \tag{28}$$

where t is the time step; $N(t)$ is the total number of vehicles arriving destinations in a period of time until t; v_i is one of the vehicles that reached destination in the time period. $T(v_i)$ is the traveling time of vehicle v_i;

Every 100 time steps, these figures are estimated, and the time period is set as 100 time steps. These two criteria can reflect the traffic condition in the road network; lower $NumV$ means less congestion happened in the road network; lower $aveT$ reflects that vehicles were guided by better routes and the time they cost on waiting in the road network is reduced. So these two criteria are adopted to evaluate whether the HSLRG is converged.

5.4. Experiment. In this part, simulations are conducted to evaluate the performance of the proposed HSLRG. In order to evaluate the performance of the proposed method, the drivers' acceptance of guidance is supposed as 100%. The updating interval of higher level network is set as 30 time steps, and the updating interval of subnetworks network is 60 time steps. The data shown in the following tables are results of the average of multiple independent simulations. In order to accelerate the converge of reinforcement learning at early stage of simulation and keep Q-values stable at middle and final stage, the learning rate α of Sarsa learning is changed depending on the time step of simulation. The concept of Simulated Annealing [36] is introduced, and the equation can be seen as follows:

$$\alpha = a * \left(1 - \frac{t}{MAXTIME}\right)^b + minimum\alpha \tag{29}$$

where t is the current time of simulation, MAXTIME is the total simulation time, a and b are constants, and minimumα is the lower limit of α.

Table 3 presents the results of Experiment 1, Experiment 2, and Experiment 3. Figures 6(a), 6(b), 6(c), 6(d), 6(e) and

input: Destination d, Network G
output: Q_d^n
begin
$Q_d^{n-1} \longleftarrow Q_d^n$
for link $s_{ij} \in L(G)$ **do**
  if $t_{ij}^d \neq null$ **then**
    $Q_d^n(i,j) \longleftarrow Q_d^{n-1}(i,j) + \alpha * (t_{ij}^d + \gamma * Q_d^{n-1}(j,k) - Q_d^{n-1}(i,j))$
  end if
end for
end

Algorithm 5: Procedure of Updating.

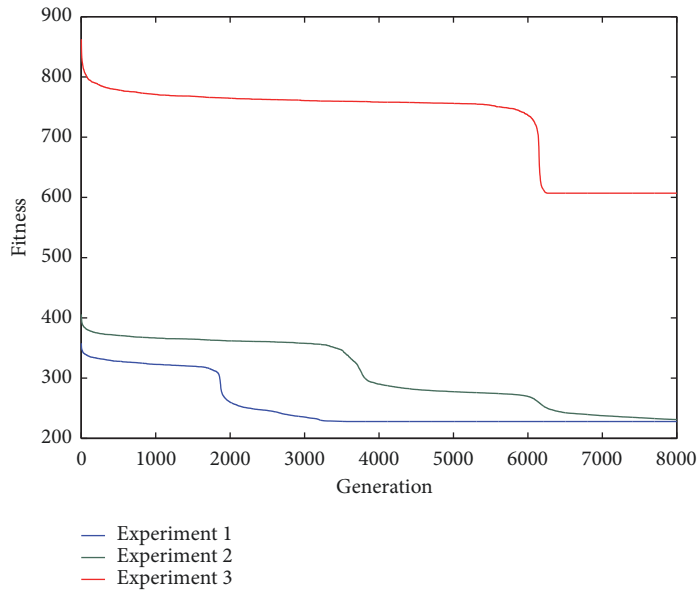
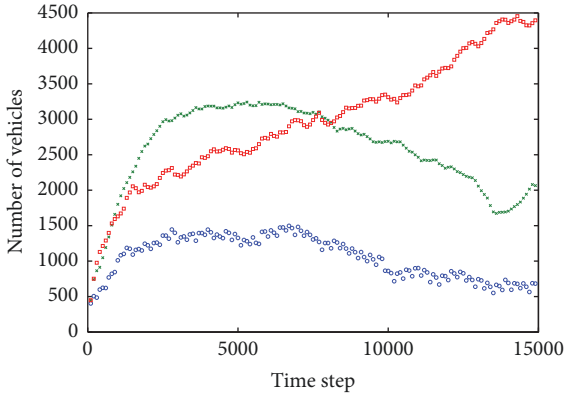


Figure 5: The evolution process of road network in Experiment 1, Experiment 2, and Experiment 3.

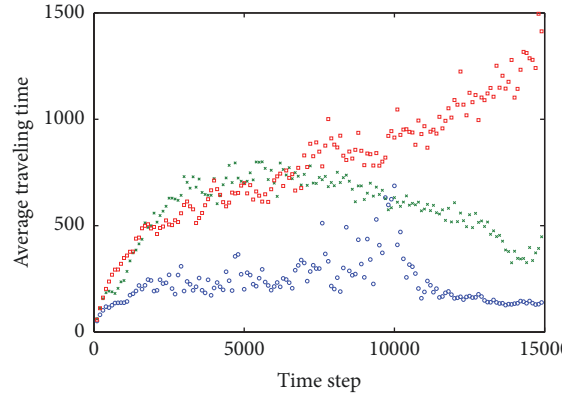
6(f) in Figure 6 show *NumV* and *aveT* of these experiments, respectively. Table 4 shows the mean and standard deviation (Std) of these experiments.

As shown in Figures 6(a)–6(f), HSLRG has lower figures of evolution values than SLWBD and DA almost during the entire simulations. These data indicate that HSLRG is fitting for guide vehicles in the large scale route network; it can alleviate the congestion phenomena and reduce the traveling time and traveling distance of vehicles in the larger scale route network. In Figures 6(a)–6(d), the tendency of *NumV* and *aveT* of HSLRG and SLWBD becomes decreasing after early stage of simulation (about 5000 time steps in Experiment 1, and about 2000 time steps in Experiment 2) while as shown in Figures 6(e) and 6(f), the evaluation values of SLWBD increased dramatically during the total 15000 time steps. The data indicate that, in limited size of road network, SLWBD has reasonable performance; however, in the larger scale road the performance of SLWBD becomes poor. As Figures 6(a)–6(f) show, the measured values of DA increased continuously. This performance indicates that DA is not a proper method for route guidance in the dynamic environments. The main reason is that DA only considers the static shortest routes, which may cause negative behavioral phenomena in dynamic transportation system, including overreaction and concentration phenomena. As shown in Table 4, from the mean and Std of *NumV* and *aveT*, we can see that the performance of proposed HSLRG dominates that of SLWBD and DA, which can prove the effectiveness of the proposed HSLRG.

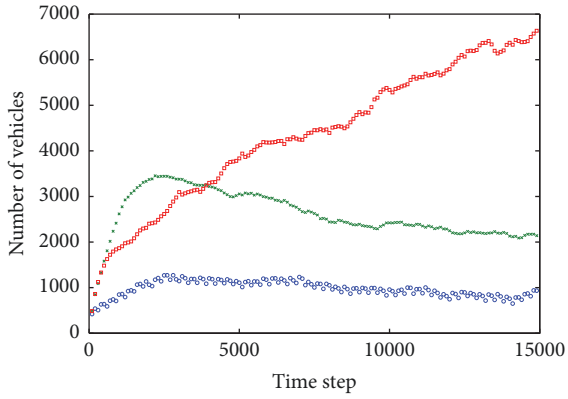
As shown in Table 3, it can be seen that in all the experiments HSLRG has the best performance and outweighs the other two methods; the statistic data indicate that vehicles guided by this algorithm have not only the largest number of vehicles arriving destinations and the least mean traveling time, but also the least traveling distance. SLWBD has better performance than DA in Experiment 1 and Experiment 2 but worse performance in Experiment 3. The statistic result indicates that Sarsa learning based route guidance on the original road network is not suitable for guiding vehicles in the large scale road network. It is because the speed of convergence of reinforcement learning depends on the scale of the searching space, and it is exponential growth



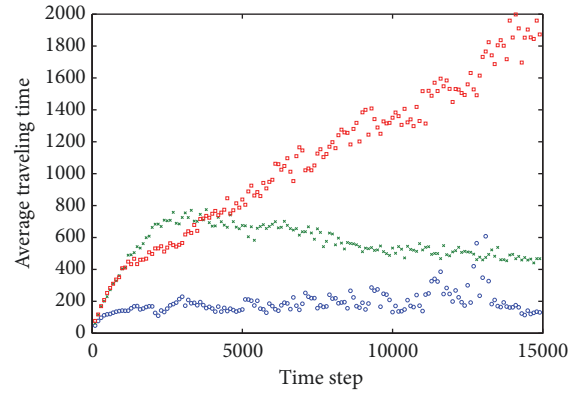
(a) The *NumV* of Experiment 1



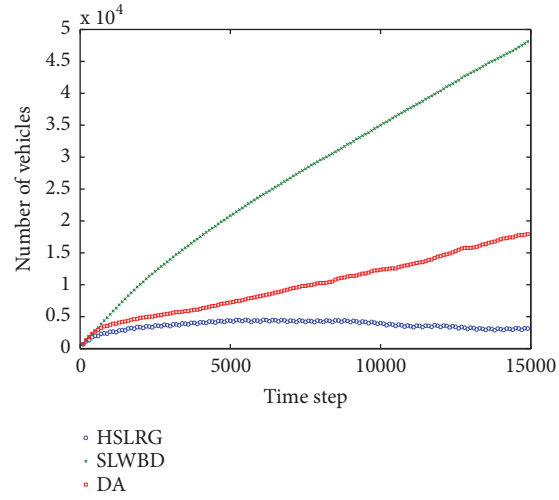
(b) The *aveT* of Experiment 1



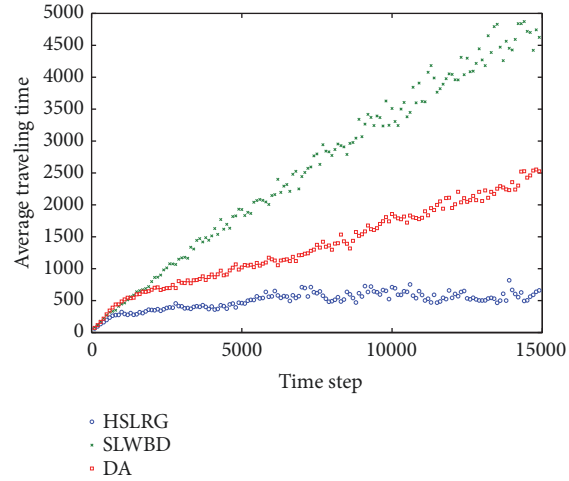
(c) The *NumV* of Experiment 2



(d) The *aveT* of Experiment 2



(e) The *NumV* of Experiment 3



(f) The *aveT* of Experiment 3

Figure 6: The results of Experiments.

Table 3: Results of Experiments.

Experiment	Algorithm	Number of arriving vehicles	Mean duration of each vehicle	Mean route length of vehicle
Experiment 1	HSLRG	63,862.3	308.03	1,880.81
	SLWBD	63,542.67	616.88	6,876.22
	DA	49,604.25	757.67	3,931.18
Experiment 2	HSLRG	70,572.33	194.81	1,639.80
	SLWBD	62,193.4	670.91	5,127.49
	DA	51,193.1	1,087.06	4,792.38
Experiment 3	HSLRG	95,919.17	456.12	4,029.92
	SLWBD	50,690.11	2,738.79	23,735.61
	DA	78,696.5	1,384.96	9,160.27

Table 4: Mean and Std of Experiment results.

Experiment	Algorithm	Mean *NumV*	Std *NumV*	Mean *aveT*	Std *aveT*
1	HSLRG	1055	296.9	236.6	108.5
	SLWBD	2578	609.8	576	163.9
	DA	2969	859.6	789	282.2
2	HSLRG	975.4	174.1	190	73.43
	SLWBD	2606	523.5	560	125.3
	DA	4347	1502	1094	477.4
3	HSLRG	3644	711.7	498.5	137.1
	SLWBD	27130	13160	2606	1382
	DA	9943	4463	1370	628.5

with the increasing of the scale of road network. And the proposed HSLRG introduced optimized Multilevel Network structure, by which route guidance on the subnetwork and route guidance on the higher level network are combined to compress the searching space of the traffic system. So, the proposed HSLRG can enhance the efficiency of CDRGS greatly.

6. Conclusion

In this paper, we have proposed the hierarchical Sarsa learning based route guidance algorithm (HSLRG) to solve route guidance problem in large scale road networks. HSLRG applies Multilevel Network method to reduce the state space of the traffic environment, which can greatly accelerate convergence of the route guidance algorithm. The effectiveness and efficiency of HSLRG were studied in three different scale road networks. The simulation results show that, in the large scale road network, comparing with SLWBD and DA, HSLRG can guide vehicles to the destinations more effectively. How to guide vehicles with multiobjective and considering personality of drivers are worthwhile for future research.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant No. 61672359 and Doctoral Scientific Research Foundation of Liaoning Province, No. 20170520197.

References

[1] A. F. Lentzakis, S. I. Ware, R. Su, and C. Wen, "Region-based prescriptive route guidance for travelers of multiple classes," *Transportation Research Part C: Emerging Technologies*, vol. 87, pp. 138–158, 2018.

[2] J. Guo and C. Liu, "Time-dependent vehicle routing of free pickup and delivery service in flight ticket sales companies based on carbon emissions," *Journal of Advanced Transportation*, vol. 2017, Article ID 1918903, 14 pages, 2017.

[3] I. S. Cases, "Minimizing the impact of large freight vehicles in the city: A multicriteria vision for route planning and type of vehicles," *Journal of Advanced Transportation*, vol. 2018, 8 pages, 2018.

[4] N. Nishi, "Evaluation for effectiveness of cdrgs (centrally determined route guidance system)," in *Proceedings of the World Congress on Intelligent Transport Systems*, 1997.

[5] N. G. Bean, F. P. Kelly, and P. G. Taylor, "Braess's paradox in a loss network," *Journal of Applied Probability*, vol. 34, no. 1, pp. 155–159, 1997.

[6] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische Mathematics*, vol. 1, no. 1, pp. 269–271, 1959.

[7] P. E. Hart, N. J. Nilsson, and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," *Systems Science & Cybernetics IEEE Transactions on*, vol. 4, no. 2, pp. 100–107, 1968.

[8] M. Papageorgiou, M. Ben-Akiva, J. Bottom et al., "Chapter 11 ITS and traffic management," in *Handbooks in Operations Research and Management Science*, C. Barnhart and G. Laporte, Eds., vol. 14, pp. 715–774, Transportation, Elsevier, 2007, http://www.sciencedirect.com/science/article/pii/S0927050706140116.

[9] D. Eppstein, "Finding the k shortest paths," in *Proceedings of the Annual IEEE Symp.on Foundations of Computer Science*, vol. 28, 2, pp. 154–165, 1994.

[10] A. Mimura, S. Sumino, and S. Kato, *Adaptive Reinforcement Learning for Dynamic Environment Based on Behavioral Habit*, Springer Berlin Heidelberg, 2012.

[11] M. Nagayoshi, H. Murao, and H. Tamaki, "Reinforcement learning for dynamic environment: A classification of dynamic environments and a detection method of environmental changes," *Artificial Life and Robotics*, vol. 18, no. 1-2, pp. 104–108, 2013.

[12] A. Chis, J. Lunden, and V. Koivunen, "Reinforcement learning-based plug-in electric vehicle charging with forecasted price," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 5, pp. 3674–3684, 2017.

[13] Y. Zou, T. Liu, D. Liu, and F. Sun, "Reinforcement learning-based real-time energy management for a hybrid tracked vehicle," *Applied Energy*, vol. 171, pp. 372–382, 2016.

[14] S. Yu, J. Zhou, B. Li, S. Mabu, and K. Hirasawa, "Q value-based Dynamic Programming with SARSA Learning for real time route guidance in large scale road networks," in *Proceedings of the International Joint Conference on Neural Networks*, pp. 1–7, 2012.

[15] F. Wen and X. Wang, "Sarsa learning based route guidance system with global and local parameter strategy," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. E98A, no. 12, pp. 2686–2693, 2015.

[16] A. G. Barto and S. Mahadevan, *Discrete Event Dynamic Systems*, vol. 13, no. 1/2, pp. 41–77.

[17] H. Ding, F. Guo, X. Zheng, and W. Zhang, "Traffic guidance–perimeter control coupled method for the congestion in a macro network," *Transportation Research Part C: Emerging Technologies*, vol. 81, pp. 300–316, 2017.

[18] A. Valejo, M. C. Ferreira de Oliveira, G. P. Filho, and A. d. Lopes, "Multilevel approach for combinatorial optimization in bipartite network," *Knowledge-Based Systems*, vol. 151, pp. 45–61, 2018.

[19] F. Wen, S. Mabu, and K. Hirasawa, "A genetic algorithm based clustering method for optimal route calculation on multilevel networks," *Sice Journal of Control Measurement System Integration*, vol. 4, pp. 83–88, 2011.

[20] S. Jung and S. Pramanik, "An efficient path computation model for hierarchically structured topographical road maps," *Knowledge & Data Engineering IEEE Transactions on*, vol. 14, no. 5, pp. 1029–1046, 2002.

[21] L. Kaufman and P. Rousseeuw, *Finding Groups in Data: An Introduction to Cluster Analysis*, DBLP, 1990.

[22] Z. Huang and M. K. Ng, "A fuzzy k-modes algorithm for clustering categorical data," *IEEE Transactions on Fuzzy Systems*, vol. 7, no. 4, pp. 446–452, 1999.

[23] B. Ji, X. Yuan, and Y. Yuan, "A binary borg-based heuristic method for solving a multi-objective lock and transshipment co-scheduling problem," *IEEE Transactions on Intelligent Transportation Systems*, vol. 99, pp. 1–12, 2018.

[24] T. Karthy and K. Ganesan, "Fuzzy multi objective transportation problem – evolutionary algorithm approach," *Journal of Physics: Conference Series*, 2018.

[25] G. Tian, Y. Ren, Y. Feng, M. Zhou, H. Zhang, and J. Tan, "Modeling and planning for dual-objective selective disassembly using AND/OR graph and discrete artificial bee colony," *IEEE Transactions on Industrial Informatics*, 2019.

[26] G. Tian, M. Zhou, and P. Li, "Disassembly sequence planning considering fuzzy component quality and varying operational cost," *IEEE Transactions on Automation Science and Engineering*, vol. 15, no. 2, pp. 748–760, 2018.

[27] R. Storn and K. Price, "Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization*, vol. 11, no. 4, pp. 341–359, 1997.

[28] K. V. Price, R. M. Storn, and J. A. Lampinen, "Differential evolution-a practical approach to global optimization," *Natural Computing*, vol. 2, pp. 393–405, 2005.

[29] G. Tian, Y. Ren, and M. Zhou, *Dual-Objective Scheduling of Rescue Vehicles to Distinguish Forest Fires via Differential Evolution and Particle Swarm Optimization Combined Algorithm*, IEEE Press, 2016.

[30] T. Yu, S. Zhang, and Y. Hong, *Dynamic Optimal CPS Control for Interconnected Power Systems Based on SARSA Algorithm*, Springer, NY, USA, 2014.

[31] N. Lilith and K. Doğançay, "Distributed reduced-state SARSA algorithm for dynamic channel allocation in cellular networks featuring traffic mobility," in *Proceedings of the IEEE International Conference on Communications*, vol. 2, pp. 860–865, 2005.

[32] T. G. Dietterich, "Hierarchical reinforcement learning with the MAXQ value function decomposition," *Journal of Artificial Intelligence Research*, vol. 13, no. 1, pp. 227–303, 2000.

[33] S. Mozer, M. C, and M. Hasselmo, "Reinforcement learning: An introduction," pp. 285–286, 2005.

[34] P. Sanders and D. Schultes, "Highway hierarchies hasten exact shortest path queries," in *Proceedings of the European Conference on Algorithms*, vol. 3669, pp. 568–579, 2005.

[35] L. Bieker, "Recent development and applications of sumo - simulation of urban mobility," *International Journal on Advances in Systems & Measurements*, vol. 3, no. 4, pp. 128–138, 2012.

[36] D. Bertsimas and J. Tsitsiklis, "Simulated annealing," *Statistical Science*, vol. 8, no. 1, pp. 10–15, 1993.