

Technical report 16-014

# **Parameterized dynamic routing of a fleet of cybercars\***

R. Luo, T.J.J. van den Boom, and B. De Schutter

*If you want to cite this report, please use the following reference instead:*

R. Luo, T.J.J. van den Boom, and B. De Schutter, "Parameterized dynamic routing of a fleet of cybercars," *Proceedings of the 14th IFAC Symposium on Control in Transportation Systems (CTS 2016)*, Istanbul, Turkey, pp. 55–60, May 2016.

Delft Center for Systems and Control  
Delft University of Technology  
Mekelweg 2, 2628 CD Delft  
The Netherlands  
phone: +31-15-278.24.73 (secretary)  
URL: <https://www.dcsc.tudelft.nl>

---

\*This report can also be downloaded via [https://pub.deschutter.info/abs/16\\_014.html](https://pub.deschutter.info/abs/16_014.html)

# Parameterized Dynamic Routing of a Fleet of Cybercars<sup>\*</sup>

Renshi Luo<sup>\*</sup> Ton J.J. van den Boom<sup>\*</sup> Bart De Schutter<sup>\*</sup>

*<sup>\*</sup>Delft Center for Systems and Control, Delft University of Technology, Delft, The Netherlands (e-mail: r.luo@tudelft.nl; a.j.j.vandenboom@tudelft.nl; b.deschutter@tudelft.nl).*

---

**Abstract:** Due to the nonlinearity of the dynamics of vehicles and the discrete nature of the route decision variables, the dynamic routing problem for a large number of vehicles is computationally very hard to solve. In this paper, two efficient parameterized control methods are proposed for the dynamic routing of a fleet of cybercars in a road network only open to cybercars. With the proposed parameterized control methods, the updates of the routes of cybercars are parameterized and then optimized over the parameters with respect to the overall performance of the cybercar system for a representative set of scenarios. After tuning the parameters, the proposed parameterized control methods are implemented online with fixed parameters. Moreover, the two proposed parameterized control methods are well-structured and scalable, and therefore can be applied to road networks with arbitrary topologies. The effectiveness of the proposed parameterized control methods is shown in a numerical simulation study.

*Keywords:* Dynamic routing; Transportation Control; Intelligent transportation systems.

---

## 1. INTRODUCTION

In recent years, the numbers of private cars in many big cities all around the world have greatly increased. Although much effort has been spent on improving the infrastructure, such as building more roads and installing more advanced traffic information systems, the highly disorganized behaviors of the human drivers are still causing severe problems, such as frequent congestion, high numbers of accidents, increasing energy consumption and pollution, and high levels of noise, etc. Due to these severe problems, the quality of life and the environment in big cities has been degraded.

Although public transportation systems (e.g., buses, trams and subways) have been considered capable of solving these problems, due to prefixed time schedules and routes, public transportation systems inherently cannot offer the same level of personal mobility to passengers as private cars. Therefore, even though public transportation systems have been continuously improved, people that are in favor of personal mobility still prefer using private cars. As a result, the problems caused by the increasing use of private cars in big cities are still largely unsolved.

A novel and promising approach for personal mobility, emerging as an alternative solution to the use of private cars, is to use a cybernetic transportation system, which is an intelligent transportation system providing on-demand and door-to-door service, see e.g., Parent and Texier (1993); Parent (1997, 2007); Naranjo et al. (2009). More specifically, a cybernetic transportation system is exclusively formed by cybercars, which are small-sized and automated vehicles typically accommodating 3 to 6 seated passengers. Cybercars have a high flexibility and reactivity, providing on-demand and door-to-door transportation service. Hence, a cybernetic transportation system offers

better urban mobility than conventional public transportation systems, see Parent (2010). Besides, cybercars are powered by electricity, which is more efficient and less polluting than fossil fuels. Actually, according to Awasthi et al. (2011), cybercars are even competitive on a per passenger-km basis compared with public transportation in terms of energy consumption. In addition, since electric motors generate much less noise than gasoline motors, using a cybernetic transportation system will greatly reduce the noise levels in the urban environment. So far, several projects, such as CyberCars see Parent et al. (2003), CyberCars2, CyberC3 see Yang et al. (2006), have been dedicated to the development and dissemination of a cybernetic transportation system.

According to Bishop (2005), the fast development of automated driving technologies, such as adaptive cruise control, automated lane change and path following, etc, has enabled individual vehicles to drive autonomously. However, due to the absence of efficient strategies for the control of a fleet of cybercars, cybernetic transportation systems are still not widely used on a large scale basis.

Actually, the fleet control problem of cybercars has already been considered in the literature. More specifically, in Awasthi et al. (2011), the problem was considered conceptually from a centralized point of view and then a centralized fleet management system of cybercars was proposed. In Berger et al. (2011), a new concept of control that merges centralized and decentralized control approaches, was proposed for the fleet control problem of cybercars. However, that paper just discussed how the new concept can help in dealing with disturbances from the environment, but it did not introduce a specific control algorithm. In our previous work, see Luo et al. (2014), we proposed a discrete-time model for the dynamics and the energy consumption of cybercars and we formulated a specific instance of the fleet control problem of cybercars, i.e., the dynamic routing problem. However, we did not propose an efficient control

---

<sup>\*</sup> This work was supported by the China Scholarship Council under Grant 201207090001.

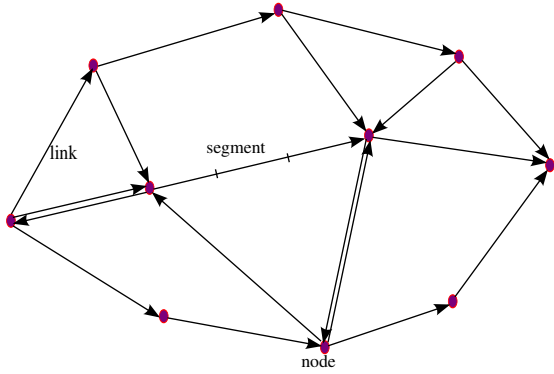


Fig. 1. Cybernetic transportation network

method to solve the problem. With respect to the literature, in this paper, we propose efficient control methods for the dynamic routing problem of cybercars.

In fact, the dynamics of cybercars are highly nonlinear and the routes of cybercars are discrete. This leads to the fact that the dynamical routing problem of a fleet of cybercars is actually a mixed integer nonlinear programming problem, which is computationally very hard to solve. For this reason, instead of computing the optimal routes of cybercars directly, we propose to use efficient control methods in which the route selection process is parameterized, and then optimize the parameters of the control methods with respect to the performance, e.g., total time spent and total energy consumption, of cybercars.

This paper is organized as follows. In Section 2, we present the general description of the dynamic routing problem of a fleet of cybercars. In Section 3, the discrete-time model of the dynamics and the energy consumption of cybercars used in this paper is described. In Section 4, two parameterized control methods are proposed for the dynamic routing of cybercars. In Section 5, a simple numerical case study is presented to demonstrate the effectiveness of the proposed control methods. Finally, in Section 6, we summarize the results of this paper and present some ideas for future work.

## 2. THE DYNAMIC ROUTING PROBLEM OF CYBERCARS

A cybernetic transportation network is a road network only open to cybercars. Conceptually, a cybernetic transportation network can be represented by a graph like the one shown in Figure 1, where each road is represented by a directed link and each intersection is represented by a node. In addition, we consider that each link is divided into a number of segments and each segment has a length typically in the range of 50 to 100 m.

We assume that there are higher-level controllers deciding schedules (including starting time, departure point and destination) for all cybercars and only focus on the dynamic routing problem of all cybercars from their departure points to their destinations. We assume that, at any time, the traffic density (i.e. the number of cybercars per unit of length) in each segment determines the speeds of all the cybercars running in that segment. We also assume each segment has a maximum capacity, which is the maximum allowed number of cybercars in the segment at the same time. Note that at any time, if the number of cybercars in a segment reaches or exceeds the maximum capacity, that

segment will be blocked and no more cybercars are allowed to enter the blocked segment. In addition, we assume that all cybercars have the same length and the same mass. Finally, we assume that within a simulation time interval, no cybercar can cover a distance longer than the length of a segment.

By taking the real-time conditions (i.e., the positions of cybercars, the number of cybercars in each segment and the blocked or unblocked state of each segment) of the network into account, we aim to update the routes of cybercars dynamically so that the overall system performance, i.e., a weighted sum of the total time spent and the total energy consumption of all cybercars in the network, is optimized.

## 3. DISCRETE-TIME MODELING

Given the current states of the cybercars and the network, a model of the dynamics of cybercars the network can be used to predict the future states of the cybercars and the network, which can then be used in the route selection process for cybercars.

When the departure time for a cybercar arrives, the cybercar will enter the network. After that, at each simulation time step, with the current states of all cybercars and the current conditions of the network given, the states of the cybercars and the conditions of the network at the next step need to be updated, and the energy consumption of cybercars at this step need to be calculated. In this paper, the discrete-time model for the dynamics and the energy consumption of cybercars presented in Luo et al. (2014) is used. Summarily, this model consists of three parts:

- update of states of a single cybercar
- update of states of the network
- computation of energy consumption of a single cybercar

Given the routes of all cybercars, the discrete-time model can be used to compute the total time spent and the total energy consumption by all cybercars in the network.

## 4. PARAMETERIZED CONTROL METHODS

The main idea of a parameterized control method is to parameterizing the control decision-making process and then tune the parameters off-line by solving an optimization problem considering the performance of the control method on a number of representative scenarios, see Oung and D'Andrea (2012) and Tarău et al. (2010). After that, for any specific scenario, the control decisions are made on-line by using the parameterized control method with fixed parameters.

In the dynamic routing of cybercars, at each control cycle, the parameterized control methods update the route of each cybercar by selecting a route using a parameterized control law from a limited set of possible routes. More specifically, a finite set of possible routes for each cybercar from its current position to its destination can be generated by using shortest route algorithms, e.g., *Dijkstra's Algorithm* presented in Dijkstra (1959). Besides, the whole network is divided into a set  $G$  of subnetworks and in different subnetworks the parameterized control law uses different values of parameters.

### 4.1 Generating a limited set of possible routes from one node to another

Given a static network with a fixed cost on each link, a shortest route algorithm is able to find a predefined number of minimal

cost routes from one node to another. More specifically, in this paper given the topology of the cybernetic transportation network, before a shortest route algorithm is called to generate the limited sets of possible routes for cybercars, the cost on each link  $j$ , which will be used by the shortest route algorithm, is determined by:

$$c_j = \alpha \frac{L_{\text{link},j}}{L_{\text{link,ave}}} + \beta \frac{1}{T_{\text{link,ave}}} \sum_{m=1}^{M_{\text{segment}}(j)} \frac{L_{m,j}}{f_{m,j}(\rho_{m,j}(k))} \quad (1)$$

where  $L_{\text{link},j}$  denotes the length of link  $j$ ,  $L_{\text{link,ave}}$  denotes the average of  $L_{\text{link},j}$  over all links,  $T_{\text{link,ave}}$  denotes the average link travel time over all links,  $L_{m,j}$  denotes the length of the  $m$ -th segment of link  $j$ ,  $M_{\text{segment}}(j)$  denotes the number of segments in link  $j$ ,  $\rho_{m,j}(k)$  denotes the current traffic density in the  $m$ -th segment of link  $j$ , the function  $f_{m,j}(\cdot)$  describe how the speeds of cybercars in the  $m$ -th segment of link  $j$  depend on the traffic density in that segment,  $\alpha$  and  $\beta$  are given constants.

Based on the fundamental flow-density curve for automated traffic presented in Bose and Ioannou (2003), one way to define  $f_{m,j}(\cdot)$  is given by

$$f_{m,j}(\rho_{m,j}(k)) = \begin{cases} v_{\text{free},m,j}, & \text{if } \rho_{m,j}(k) \leq \rho_{\text{crit},m,j} \\ \frac{1}{h_{\text{con},m,j}} \left( \frac{1}{\rho_{m,j}(k)} - L_{\text{veh}} \right), & \text{if } \rho_{m,j}(k) > \rho_{\text{crit},m,j} \end{cases} \quad (2)$$

where  $v_{\text{free},m,j}$  and  $h_{\text{con},m,j}$  respectively denote the free flow speed and the constant time headway of cybercars in the  $m$ -th segment of link  $j$ ,  $L_{\text{veh}}$  denotes the length of a cybercar, and  $\rho_{\text{crit},m,j}$  denotes the critical traffic density of the  $m$ -th segment of link  $j$ , which is given by

$$\rho_{\text{crit},m,j} = \frac{1}{h_{\text{con}} v_{\text{free},m,j} + L_{\text{veh}}} \quad (3)$$

The way to determine  $T_{\text{link,ave}}$  is given by:

$$\bar{v}_{\text{free}} = \frac{\sum_{j=1}^{M_{\text{link}}} \sum_{m=1}^{M_{\text{segment}}(j)} v_{\text{free},m,j}}{\sum_{j=1}^{M_{\text{link}}} M_{\text{segment}}(j)} \quad (4)$$

$$T_{\text{link,ave}} = \frac{L_{\text{link,ave}}}{\bar{v}_{\text{free}}/2} \quad (5)$$

where  $M_{\text{link}}$  denotes the number of links in the network. Note that  $\bar{v}_{\text{free}}$  represents the average free flow speed over all segments in all links, and we assume that the average speed of a cybercar is half of  $\bar{v}_{\text{free}}$ .

#### 4.2 Parameterized control method 1

We define  $R_i(k)$  as the limited set of possible routes of cybercar  $i$  generated at time  $kT$ , as described in Section 4.1. After that, for each  $r \in R_i(k)$ , we define  $L_{\text{route}}(r)$  as the length of route  $r$ ,  $T_{\text{route}}(r)$  as the estimated travel time on route  $r$ , and  $N_{\text{route}}(r)$  as the estimated number of cybercars on route  $r$ . Note that for the time-dependent variables such as  $T_{\text{route}}(r)$  and  $N_{\text{route}}(r)$ , the index  $k$  is dropped for the sake of simplicity.

Since the length of each link is fixed, the length of the route  $r$  can be easily calculated by summing up of the lengths of all the links belonging to route  $r$ . However, even if a route  $r$  is given, the travel time and the number of cybercars on that route are still time-dependent. Therefore, at any time when  $T_{\text{route}}(r)$  and  $N_{\text{route}}(r)$  are used, they have to be calculated based on the current states of all cybercars and of the network.

In this paper, we propose two approaches to estimate  $T_{\text{route}}(r)$  and  $N_{\text{route}}(r)$ . More specifically, at time step  $k$ , given the current states of all cybercars and of the network, for each  $r \in R_i(k)$ ,  $T_{\text{route}}(r)$  and  $N_{\text{route}}(r)$  are estimated as follows:

- Approach 1: Only use the current state of the network:

$$T_{\text{route}}(r) = \sum_{j \in r} \sum_{m=1}^{M_{\text{segment}}(j)} \frac{L_{m,j}}{f_{m,j}(\rho_{m,j}(k))} \quad (6)$$

$$N_{\text{route}}(r) = \sum_{j \in r} \sum_{m=1}^{M_{\text{segment}}(j)} N_{m,j}(k) \quad (7)$$

where  $\rho_{m,j}(k)$  and  $N_{m,j}(k)$  respectively denote the traffic density and number of cybercars in the  $m$ -th segment of link  $j$  at time step  $k$ .

- Approach 2: Predict the future states of the network assuming all cybercars follow the current routes and using the current states of the cybercars and the work and the simulation model:

$$\bar{\rho}_{m,j} = \sum_{l=1}^{N_p} \frac{\rho_{m,j}(k+l)}{N_p} \quad (8)$$

$$T_{\text{route}}(r) = \sum_{j \in r} \sum_{m=1}^{M_{\text{segment}}(j)} \frac{L_{m,j}}{f_{m,j}(\bar{\rho}_{m,j})} \quad (9)$$

$$N_{\text{route}}(r) = \sum_{j \in r} \sum_{m=1}^{M_{\text{segment}}(j)} \sum_{l=1}^{N_p} \frac{N_{m,j}(k+l)}{N_p} \quad (10)$$

where  $N_p$  denotes the prediction horizon,  $\bar{\rho}_{m,j}$  denotes the average traffic density on the  $m$ -th segment of link  $j$  over the prediction horizon.

Next, at time step  $k$ , for each cybercar  $i$  in subnetwork  $g \in G$ , we define such a function for each  $r \in R_i(k)$ :

$$\begin{aligned} \phi_i(r; \theta_g) = & \theta_{g,1} \cdot \frac{L_{\text{route}}(r)}{L_{\text{route,ave},i}} + \theta_{g,2} \cdot \frac{T_{\text{route}}(r)}{T_{\text{route,ave},i}} + \\ & + \theta_{g,3} \cdot \frac{N_{\text{route}}(r)}{N_{\text{route,ave},i} + \kappa} \end{aligned} \quad (11)$$

where  $\theta_{g,1}$ ,  $\theta_{g,2}$  and  $\theta_{g,3}$  are the parameters for subnetwork  $g$  and  $L_{\text{route,ave},i}$ ,  $T_{\text{route,ave},i}$ , and  $N_{\text{route,ave},i}$  are respectively the average of  $L_{\text{route}}(r)$ ,  $T_{\text{route}}(r)$ , and  $N_{\text{route}}(r)$  over all  $r \in R_i(k)$  for cybercar  $i$ , and  $\kappa$  is a small positive number added to the denominator of the last term in (11) to prevent division by 0.

Finally, based on (11), the route of each cybercar  $i$  in subnetwork  $g$  at time step  $k$  is selected as:

$$r^* = \arg \min_{r \in R_i(k)} \phi_i(r; \theta_g) \quad (12)$$

where  $\theta_g = [\theta_{g,1} \ \theta_{g,2} \ \theta_{g,3}]^T$ .

#### 4.3 Parameterized control method 2

In this method, we first define  $H_n$  as the set of outgoing links from node  $n$  and  $R_{n,d}(k)$  as the limited set of possible routes from node  $n$  to node  $d$  generated at time  $kT$ , as described in Section 4.1. After that, for each  $j \in H_n$ , we define  $L_j$  as the length of link  $j$ , and  $T_{\text{link},j}$  and  $N_{\text{link},j}$  as the estimated travel time and estimated number of cybercars on link  $j$ , respectively. Note that for the time-dependent variables such as  $T_{\text{link},j}$  and  $N_{\text{link},j}$ , the index  $k$  is dropped for the sake of simplicity.

We propose two approaches to estimate  $T_{\text{link},j}$  and  $N_{\text{link},j}$ . More specifically, at time step  $k$ , given the current states of all

cybercars and the current conditions of the network,  $T_{\text{link},j}$  and  $N_{\text{link},j}$  are estimated as follows:

- Approach 1: Only use the current state of the network:

$$T_{\text{link},j} = \sum_{m=1}^{M_{\text{segment}}(j)} \frac{L_{m,j}}{f_{m,j}(\rho_{m,j}(k))} \quad (13)$$

$$N_{\text{link},j} = \sum_{m=1}^{M_{\text{segment}}(j)} N_{m,j}(k) \quad (14)$$

- Approach 2: Predict the future states of the network assuming all cybercars follow the current routes and using the current states of the cybercars and the work and the simulation model:

$$\bar{\rho}_{m,j} = \sum_{l=1}^{N_p} \frac{\rho_{m,j}(k+l)}{N_p} \quad (15)$$

$$T_{\text{link},j} = \sum_{m=1}^{M_{\text{segment}}(j)} \frac{L_{m,j}}{f_{m,j}(\bar{\rho}_{m,j})} \quad (16)$$

$$N_{\text{link},j} = \sum_{m=1}^{M_{\text{segment}}(j)} \sum_{l=1}^{N_p} \frac{N_{m,j}(k+l)}{N_p} \quad (17)$$

Next, for each  $j \in H_n$ , we define  $\tilde{r}_{j,d_i}$  as the shortest time route<sup>1</sup> from the end of link  $j$  to the destination node  $d_i$ . After that, for each cybercar  $i$  in an incoming link of node  $n$  with the link in subnetwork  $g$ , we define the following function:

$$\begin{aligned} \varphi_{i,n}(j, \theta_g) = & \theta_{g,1} \cdot \frac{L_{\text{link},j} + L_{\text{route}}(\tilde{r}_{j,d_i})}{L_{\text{ave},n,d_i}} \\ & + \theta_{g,2} \cdot \frac{T_{\text{link},j} + T_{\text{route}}(\tilde{r}_{j,d_i})}{T_{\text{ave},n,d_i}} \\ & + \theta_{g,3} \cdot \frac{N_{\text{link},j} + N_{\text{route}}(\tilde{r}_{j,d_i})}{N_{\text{ave},n,d_i} + \kappa} \end{aligned} \quad (18)$$

where  $d_i$  denotes the destination of cybercar  $i$ ,  $\theta_{g,1}$ ,  $\theta_{g,2}$  and  $\theta_{g,3}$  are parameters for subnetwork  $g$ , and  $L_{\text{ave},n,d_i}$ ,  $T_{\text{ave},n,d_i}$ , and  $N_{\text{ave},n,d_i}$  are respectively the average of  $L_{\text{route}}(r)$ ,  $T_{\text{route}}(r)$ , and  $N_{\text{route}}(r)$  over all  $r \in R_{n,d_i}(k)$ .

Finally, the route of each cybercar  $i$  in an incoming link of node  $n$  and in subnetwork  $g$  is selected as follows:

- selecting the outgoing link from node  $n$  as:

$$j^* = \arg \min_{j \in H_n} \varphi_{i,n}(j, \theta_g) \quad (19)$$

where  $\theta_g = [\theta_{g,1} \ \theta_{g,2} \ \theta_{g,3}]^T$ .

- the entire route of cybercar  $i$  from node  $n$  to its destination  $d_i$  is selected as:

$$r^* = \{j^*\} \cup \{\tilde{r}_{j^*,d_i}\} \quad (20)$$

which means the updated route of cybercar  $i$  consists of the outgoing link  $j^*$  of node  $n$  and the shortest time route  $\tilde{r}_{j^*,d_i}$  from the end of  $j^*$  to the destination  $d_i$  of cybercar  $i$ .

#### 4.4 Tuning the parameters for parameterized control methods

After parameterizing the route selection process, we need to tune the parameters of the proposed parameterized control methods. This is done as follows.

<sup>1</sup> A shortest time route from one node to another can be found by using the link cost equation (1) with  $\alpha = 0$  for each link.

Table 1. Flows of cybercars that are allowed to update routes

Flow index	Origin node	Destination node	Percentage
1	5	7	40%
2	1	10	15%
3	8	3	15%
4	2	11	15%
5	9	4	15%

Firstly, given the starting times, the origins and the destinations of all cybercars, the overall of performance of a cybernetic transportation system is defined as

$$J = w_1 \cdot \frac{J_{\text{TTS}}}{J_{\text{TTS,typical}}} + w_2 \cdot \frac{J_{\text{TEC}}}{J_{\text{TEC,typical}}} \quad (21)$$

where  $J_{\text{TTS}}$  and  $J_{\text{TEC}}$  respectively denote the total time spent and the total energy consumption by all cybercars,  $J_{\text{TTS,typical}}$  and  $J_{\text{TEC,typical}}$  respectively denote the typical values<sup>2</sup> of the total time spent and the total energy consumption by all cybercars in one simulated period,  $w_1$  and  $w_2$  are weights.

After that, we define a scenario as a case where the starting times, the origins and the destinations of all cybercars are given. Then, the performance of a parameterized control method on a specific scenario of the dynamic routing of cybercars is evaluated by

$$J_o(\theta) = w_1 \cdot \frac{J_{\text{TTS},o}}{J_{\text{TTS,typical}}} + w_2 \cdot \frac{J_{\text{TEC},o}}{J_{\text{TEC,typical}}} \quad (22)$$

where  $o$  denotes the index of the scenario, and  $\theta = [\theta_1^T \ \theta_2^T \ \dots]^T$ .

Finally, given a number  $N^{\text{scenario}}$  of representative scenarios, the parameters  $\theta$  of the parameterized control method are tuned by minimizing the sum of  $J_o(\theta)$  over the representative scenarios. More specifically, the parameters  $\theta$  are tuned by solving the following nonlinear programming (NLP) problem:

$$\begin{aligned} \min_{\theta} \quad & \sum_{o=1}^{N^{\text{scenario}}} J_o(\theta) \\ \text{s.t.} \quad & \text{model equations} \end{aligned} \quad (23)$$

which is nonconvex and can be solved by multi runs of numerical algorithms, e.g. genetic algorithm, simulated annealing, pattern search and sequential quadratic programming, see Bertsekas (1999).

## 5. SIMULATION STUDY

We now illustrate the proposed control methods using a simulation study. In the simulations, we consider the network shown in Figure 2, where there are 3 subnetworks, 11 nodes and 18 links. Each link is 200 meters long and has 4 segments, with each segment 50 meters long. We generated 30 scenarios, of which 20 are used for tuning the parameters of the proposed control methods and the other 10 are used for evaluating the performance of the proposed control methods.

For every scenario, we set a number  $N_{\text{car,enabled}}$  of cybercars that are allowed to update routes and generate a random number  $N_{\text{car,fixed}}$  of cybercars with fixed routes. After that, the total number of cybercars are divided into 9 flows, which are summarized in Table 1 and Table 2.

<sup>2</sup> These values are e.g., the values of total time spent and total energy consumption of all cybercars in a numerical simulation where the routes of all cybercars are fixed or a simple route control strategy (e.g., fastest route) is used.

Table 2. Flows of cybercars with fixed routes

Flow index	Origin node	Destination node	Percentage
6	2	3	20%
7	9	10	20%
8	2	10	30%
9	9	3	30%

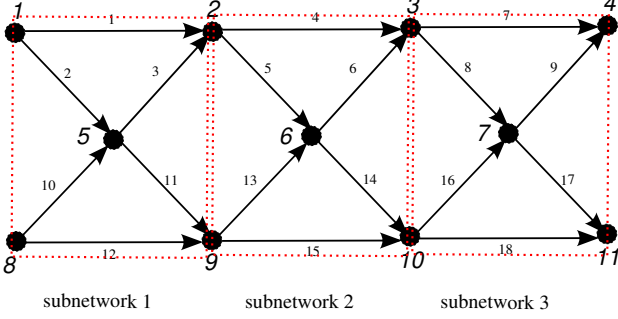


Fig. 2. Road network used in simulation study

As routing control only starts playing a role if the network is fully loaded, we define the departure times of cybercars for each of the origin-destination flows as follows. For each of the flows 1, 2 and 3, the departure time of the first cybercar is a random number uniformly distributed in the interval  $[0.8, 1.6]$ . In order to create congestion we let the first cybercar of flows 4 and 5 depart later than that of flows 1, 2, and 3, by adding an offset of 40 s. So the departure time of the first cybercar in each of the flows 4 and 5 is  $40 + a$ , where  $a$  is a random number uniformly distributed in  $[0.8, 1.6]$ . For the subsequent cybercars in flows 1 to 5 the time interval between the departure times of two consecutive cybercars is a random number uniformly distributed in  $[0.8, 1.6]$ .

Besides, for each flow of cybercars with fixed routes, the departure time of the first cybercar is  $1.2 + b$ , where  $b$  a random number uniformly distributed in  $[10, 20]$ . After that, the time interval between the departure times of two consecutive cybercars is 1.2 s.

The other parameters used in the simulations are:  $T = 1$  s,  $w_1 = 0.7$ ,  $w_2 = 0.3$ ,  $N_p = 20$ ,  $J_{TTS, \text{typical}} = 73202$  s and  $J_{TEC, \text{typical}} = 11.6767$  kWh,  $v_{\text{free}, m, j} = 60$  km/h for all  $m$  and all  $j$ , the length of each cybercar  $L_{\text{veh}} = 3.2$  m, the mass of each cybercar is  $M = 1000$  kg, the efficiency of the electric motor  $\eta_{\text{motor}} = 0.85$ , the round-trip energy recovery coefficient of the electric motor is  $\gamma_{\text{recover}} = 0.38$ . Besides, the time interval between two consecutive control steps is  $T_c = 20$  s. The simulations are performed using Matlab 2013b on a cluster computer consisting of 4 blades with 2 eight-core E5-2643 processors, and 3.3 GHz clock rate and 64 GiB memory per blade. We tuned the parameters for the two proposed parameterized control methods for both two different approaches for estimating the travel time and the number of cybercars on a route.

For simplicity of representation, we refer to the proposed parameterized dynamic routing methods as follows:

- C1: Parameterized control method 1 with Approach 1
- C2: Parameterized control method 1 with Approach 2
- C3: Parameterized control method 2 with Approach 1
- C4: Parameterized control method 2 with Approach 2

For tuning the parameters of each of the four combinations, we run the solver *fmincon* of the Matlab Optimization Toolbox with

Table 3. Scenarios for tuning the parameters

Scenario index	$N_{\text{car,enabled}}$	$N_{\text{car,fixed}}$
1	200	155
2	200	170
3	215	117
4	215	192
5	230	100
6	230	171
7	245	149
8	245	189
9	260	134
10	260	109
11	275	109
12	275	107
13	290	183
14	290	101
15	305	140
16	305	156
17	320	170
18	320	149
19	335	129
20	335	153

Table 4. CPU times (s) for tuning the parameters

C1	C2	C3	C4
93838	107510	122890	143540

Table 5. Scenarios for evaluating the performance of the parameterized control methods

Testing scenario index	$N_{\text{car,enabled}}$	$N_{\text{car,fixed}}$
t1	200	146
t2	215	184
t3	230	120
t4	245	174
t5	260	109
t6	275	168
t7	290	117
t8	305	122
t9	320	119
t10	335	142

the interior point algorithm 60 times and use a random starting point each time to solve the nonlinear programming problem (23). The 20 scenarios used for tuning the parameters of C1, C2, C3, and C4 are summarized in Table 3. Besides, the CPU times for tuning parameters are given in Table 4.

After tuning the parameters, we evaluate the performance of C1, C2, C3, and C4 on 10 different testing scenarios summarized in Table 5. Note that these 10 testing scenarios are different from the 20 scenarios used to tune the parameters of the proposed parameterized control methods.

In order to show the effectiveness of the proposed parameterized control methods, we compare the performance of the proposed parameterized control methods on the testing scenarios summarized in Table 5 with those of the following three greedy control methods using *Dijkstra's Algorithm*:

- G1: shortest distance routing method, which uses (1) with  $\alpha = 1$  and  $\beta = 0$ .
- G2: shortest time routing method, which uses (1) with  $\alpha = 0$  and  $\beta = 1$ .
- G3: combined distance and time routing method, which uses (1) with  $\alpha = 0.3$  and  $\beta = 0.7$ .

Table 6. Average online computation times (s) of the control methods

C1	C2	C3	C4	G1	G2	G3
0.5102	1.7339	2.2240	3.7941	0.1410	0.0625	0.0403

Table 7. Performance improvements of the other control methods with respect to G3, where a positive number indicates better performance

Scenario	C1	C2	C3	C4	G1	G2
t1	-4.04%	-3.48%	-6.30%	-3.13%	-3.60%	-7.46%
t2	-1.77%	-1.39%	-0.98%	-2.17%	-7.17%	-2.89%
t3	7.26%	4.27%	6.12%	4.27%	-3.43%	1.97%
t4	2.55%	8.21%	1.50%	4.97%	1.34%	-0.72%
t5	2.26%	10.34%	2.27%	3.49%	-6.30%	-1.45%
t6	3.53%	3.48%	2.37%	5.96%	-6.40%	-0.18%
t7	-0.01%	0.81%	0.20%	-2.42%	-9.70%	-4.38%
t8	-0.28%	2.84%	-0.76%	1.40%	-7.80%	-0.36%
t9	1.19%	5.27%	-0.21%	5.66%	-2.83%	-0.54%
t10	1.86%	1.64%	1.72%	-2.66%	-10.74%	-0.35%
Average	1.29%	3.20%	0.59%	1.54%	-5.66%	-1.50%

The average online computation times of all control methods over the testing scenarios are given in Table 6.

Since G3 has the best performance among all the greedy control methods, we use G3 as the benchmark and calculate the performance improvement of the other control methods on the testing scenarios. More specifically, the performance improvement of a routing method on a specific scenario compared with that of G3 on the same scenario is given by

$$\text{performance improvement} = \frac{J_{G3} - J}{J_{G3}}$$

The results of the comparison are summarized in Table 7. Note that a positive number in a cell of Table 7 indicates that the corresponding control method performs better than G3 on the corresponding scenario and vice versa.

It is seen from the simulation results that on the average, the proposed parameterized control methods perform better on the testing scenarios than the greedy control methods. Besides, the parameterized control methods that use predicted states (i.e., using estimation approach 2) of the network perform better than those use only the current states of the network at each control step. In addition, C2 performs best on the testing scenarios with better performance on 8 out of the 10 testing scenarios compared with G3. Finally, C2 has an average performance improvement of 3.20% over all the testing scenarios compared with G3.

## 6. CONCLUSIONS AND FUTURE WORK

In this paper, we have addressed the dynamic routing problem of a fleet of cybercars through a cybernetic transportation network that is only open to cybercars. In particular, based on a discrete-time model of the dynamics of the cybercars and of the network, we proposed two efficient, well-structured and scalable parameterized control methods to solve the problem such that the system performance combining the total time spent and the total energy consumption by the cybercars is optimized. Besides, we have assessed the performance of the proposed parameterized control methods experimentally by comparing them with those of greedy control methods on testing scenarios in simulation study. Simulation results highlight the effectiveness of the parameterized control methods.

In our future work, we will compare the performance of the proposed parameterized control methods with those of centralized MPC and distributed MPC.

## REFERENCES

- Awasthi, A., Chuanhan, S.S., Parent, M., and Proth, J.M. (2011). Centralized fleet management system for cybernetic transportation. *Expert Systems with Applications*, 38(4), 3710–3717.
- Berger, T., Sallez, Y., Raileanu, S., Tahon, C., Trentesaux, D., and Borangiu, T. (2011). Personal rapid transit in an open-control framework. *Computer & Industrial Engineering*, 61(2), 300–312.
- Bertsekas, D.P. (1999). *Nonlinear Programming*. Athena Scientific.
- Bishop, R. (2005). *Intelligent Vehicles Technology and Trends*. Artech House.
- Bose, A. and Ioannou, P. (2003). Mixed manual/semi-automated traffic: a macroscopic analysis. *Transportation Research Part C: Emerging Technologies*, 11(6), 439–462.
- Dijkstra, E.W. (1959). A note on two problems in connection with graphs. *Numerische Mathematik*, 1(1), 269–271.
- Luo, R., van den Boom, T., and De Schutter, B. (2014). Modeling of the dynamics and the energy consumption of a fleet of cybercars. In *2014 European Control Conference (ECC)*, 720–725. Strasbourg, France.
- Naranjo, J., Bouraoui, L., García, R., Parent, M., and Sotelo, M. (2009). Interoperable control architecture for cybercars and dual-mode cars. *IEEE Transactions on Intelligent Transportation Systems*, 10(1), 146–154.
- Oung, R. and Cruz, M.P. and D’Andrea, R. (2012). A parameterized control methodology for a modular flying vehicle. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 532–538. Algarve, Portugal.
- Parent, M. (1997). Automated public vehicles: A first step towards the automated highway. In *Proceedings of 4th World Congress on Intelligent Transport Systems*. Berlin, Germany.
- Parent, M. (2007). Advanced urban transport: Automation is on the way. *IEEE Intelligent Systems*, 22(2), 9–11.
- Parent, M. (2010). Cybercars for sustainable urban mobility - A European collaborative approach. *SAE International Journal of Passenger Cars-Electronic and Electrical Systems*, 3(2), 220–223.
- Parent, M., Gallais, G., Alessandrini, A., and Chanard, T. (2003). Cybercars: Review of first projects. In *Proceedings of 9th International Conference on Automated People Movers*. Singapore.
- Parent, M. and Texier, P. (1993). A public transport system based on light electric cars. In *Proceedings of 4th International Conference on Automated People Movers*, 154–161. Irving, USA.
- Taräu, A., De Schutter, B., and Hellendoorn, J. (2010). Model-based control for route choice in automated baggage handling systems. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, 40(3), 341–351.
- Yang, M., Wang, C., Yang, R., and Parent, M. (2006). Cyberc3: Cybercars automated vehicles in china. In *Proceedings of Transportation Research Board Annual Meeting*.