

## Multi-objective optimal predictive control of signals in urban traffic network

Xiang Li & Jian-Qiao Sun


To cite this article: Xiang Li & Jian-Qiao Sun (2019): Multi-objective optimal predictive control of signals in urban traffic network, Journal of Intelligent Transportation Systems, DOI: [10.1080/15472450.2018.1504294](https://doi.org/10.1080/15472450.2018.1504294)

To link to this article: <https://doi.org/10.1080/15472450.2018.1504294>



Published online: 15 Jan 2019.



Submit your article to this journal 



Article views: 5



View Crossmark data 

# Multi-objective optimal predictive control of signals in urban traffic network

Xiang Li<sup>a,b</sup> and Jian-Qiao Sun<sup>c</sup>

<sup>a</sup>College of Sciences, Northeastern University, Shenyang, China; <sup>b</sup>Key Laboratory of Vibration and Control of Aero-Propulsion System Ministry of Education, Northeastern University, Shenyang, China; <sup>c</sup>School of Engineering, University of California, Merced, CA, USA

## ABSTRACT

Traffic congestion in urban network has been a serious problem for decades. In this paper, a novel dynamic multi-objective optimization method for designing predictive controls of network signals is proposed. The popular cell transmission model (CTM) is used for traffic prediction. Two network models are considered, i.e., simple network which captures basic macroscopic traffic characteristics and advanced network that further considers vehicle turning and different traveling routes between origins and destinations. A network signal predictive control algorithm is developed for online multi-objective optimization. A variety of objectives are considered such as system throughput, vehicle delay, intersection crossing volume, and spillbacks. The genetic algorithm (GA) is applied to solve the optimization problem. Three example networks with different complexities are studied. It is observed that the optimal traffic performance can be achieved by the dynamic control in different situations. The influence of the objective selection on short-term and long-term network benefits is studied. With the help of parallel computing, the proposed method can be implemented in real time and is promising to improve the performance of real traffic network.

## ARTICLE HISTORY

Received 17 May 2017  
Revised 20 July 2018  
Accepted 22 July 2018

## KEYWORDS

Multi-objective optimization;  
dynamic predictive control;  
traffic network; cell  
transmission model;  
genetic algorithm

*2010 Mathematics Subject  
Classification. Primary*

## 1. Introduction

In order to relieve traffic network congestion especially in large cities, efficient control strategy has always been strongly demanded. This paper proposes a dynamic multi-objective optimization method for designing predictive controls of signals to improve traffic performance in urban network.

Network signal controls have been implemented in many cities for a couple of decades. A number of control design methods have been developed (Robertson & Bretherton, 1991; Li & Sun 2018a; Ghanim & Abu-Lebdeh, 2015; Zhang & Chang, 2014; Olia, Abdelgawad, Abdulhai, & Razavi, 2017; Hale et al., 2017). Lo and Chow (2004) applied the dynamic intersection signal control optimization (DISCO) to a congested network in Hong Kong and studied the various control strategies enabled by DISCO. Aboudolas, Papageorgiou, and Kosmatopoulos (2009) presented a framework for real-time network signal control in large-scale urban traffic networks, which combines store-and-forward traffic flow modeling, mathematical optimization, and optimal control. Yang and Jayakrishnan (2015) developed a real-time network-wide traffic signal control scheme that is flexible in response to variations

of traffic flows and can handle the expected route flows in the terms of long-term green time ratios for intersection movement. Diakaki, Papageorgiou, and Aboudolas (2002) proposed a traffic-responsive urban control (TUC) strategy using the linear-quadratic regulator. The feedback regulator of TUC is designed offline and controls the traffic signals online with real-time traffic measurements. Prashanth and Bhatnagar (2011) proposed a reinforcement learning algorithm with function approximation for traffic signal control. The algorithm incorporates state-action features and is easily implementable in high-dimensional settings. Chow and Lo (2007) developed a novel sensitivity analysis of signal control with physical queues. The effects of physical queuing are taken into account, including queue spillback and blockage. The proposed methodology shows promising numerical results and can be applied in dynamic signal control, traffic assignment, and transport network design. Network signal control was investigated by Dotoli, Fanti, and Meloni (2006) using a discrete time model that is embedded in a real-time control. The control determines the signal timing subject to technical, physical, and operational constraints.

Papatzikou and Stathopoulos (2015) combined dynamic traffic assignment and network control to

minimize the potential loss induced to travelers. Liu, Han, Gayah, Friesz, and Yao (2015) proposed a two-stage online signal control strategy for dynamic networks using a linear decision rule approach and a distributionally robust optimization technique. A reinforcement learning-based algorithm was proposed by Ozan, Baskan, Haldenbilen, and Ceylan (2015) to find optimal signal timings in coordinated signalized networks for a fixed set of link flows. The application of meta-heuristic optimization methods to network signal setting was investigated by Cantarella, de Luca, Di Pace, and Memoli (2015), where a new traffic flow model combining cell transmission model (CTM) and platoon dispersion model is used. A computationally efficient simulation-based optimization framework was improved by Chen, Xiong, He, Zhu, and Zhang (2016) to optimize dynamic traffic assignment.

Along with the development of traffic network models with high accuracy and extensibility, simulation-based optimization strategies have been widely used (Li & Sun, 2015, 2017). Model predictive control (MPC), which is an optimal control method applied in a rolling horizon framework, appears to be very promising and its key ideas are adopted in this study. MPC has the following advantages compared with other control strategies (Hegyi, Schutter, & Hellendoorn, 2005). MPC operates in closed-loop with the instant traffic demands as inputs. They are robust to uncertainty, disturbances, and model mismatch. The future of the traffic system is predicted with the updated model, which in turn helps to determine a reliable control. A short prediction horizon is usually sufficient, which reduces complexity and makes the real-time application feasible.

Lin, De Schutter, Yugeng, and Hellendoorn (2011) applied MPC to control and coordinate urban traffic networks. The online optimization problem is formulated into a mixed-integer linear programming (MILP) problem to increase the real-time feasibility of MPC. An optimal perimeter control for two-region urban cities was formulated by Geroliminis, Haddad, and Ramezani (2013) and solved by implementing the MPC scheme with the use of macroscopic fundamental diagram. A framework for multi-agent control of linear dynamic systems was proposed by Oliveira and Camponogara (2010). The authors decomposed a centralized MPC problem into a network of coupled subproblems that are solved by the distributed agents. Hegyi et al. (2005) applied online control to optimally coordinate variable speed limits with the aim of suppressing shock waves. The objective was to find the control signals that minimize the total time of

vehicles staying in the network. Model-based controls for road network optimization in  $L^2$ -norm and  $L^\infty$ -norm were designed by Lin, Zhou, and Xi (2013).

The studies in the literature on network signal control mostly focus on improving a single objective. Research can be hardly found on multiple objective optimizations with online predictive control. This paper proposes a dynamic multi-objective optimization method for designing real-time signal control in urban traffic network. The CTM is used for traffic prediction. Example networks with different complexities and scales are considered to show the adaptability and extensibility of the proposed method. The genetic algorithm (GA) is adopted to solve the multi-objective optimization (MOP) problem. The performance objectives such as maximizing system throughput, reducing traffic delay, and avoiding spillbacks are considered in this study. A Pareto set of optimal controls can be derived online. An algorithm is proposed to assist the user to select and implement the optimal designs.

The paper is organized as follows. The traffic network model is presented in Section 2. Section 3 presents the design of dynamic signal predictive control. The network MOP is formulated in Section 4. Numerical optimization results of case studies and performance analysis of the control are presented in Section 5. We close the paper with conclusions in Section 6.

## 2. Model description

### 2.1. Cell transmission model

The CTM initially proposed by Daganzo (1994, 1995) to describe the traffic flow on highways can be used to model urban traffic. It is a discretized form of the macroscopic hydrodynamic Lighthill, Whitham, Richards (LWR) model (Lighthill & Whitham, 1955; Richards, 1956) and is based on a simplified trapezoidal fundamental diagram. It captures the macroscopic features of the traffic including shockwaves, queue formation, and queue dissipation in both congested and uncongested situations.

Each road section is modeled as a sequence of cells whose lengths are the products of free flow speed on that link  $v_{free}$  (m/s) and duration of a time step  $\Delta t$  (s). The maximum number  $N_i$  of vehicles that can be contained in cell  $i$  is the product of cell length  $l$  (m) and jam density  $k_{jam}$  (veh/m).

The dynamics in the CTM is governed by the following two equations at each time step  $t$  for cell  $i$ . The outflow  $q_i(t)$  (veh/ $\Delta t$ ) of cell  $i$ , which is also the inflow of cell  $i+1$  at time step  $t$ , is given by,

$$q_i(t) = \min \left\{ n_i(t)/\Delta t, Q_i, \omega(N_{i+1} - n_{i+1}(t))/(v_{free}\Delta t) \right\}, \quad (2.1)$$

where  $n_i(t)$  denotes the number of the existing vehicles in cell  $i$  at the beginning of time step  $t$ , and  $Q_i$  (veh/ $\Delta t$ ) represents the outflow capacity of cell  $i$ . The outflow is also limited to the available space in the following cell, multiplied by the ratio of backward wave speed  $\omega$  (m/s) and  $v_{free}$ .

After all the flows between cells have been determined, the number of vehicles  $n_i(t)$  in each cell can be updated as,

$$n_i(t+1) = n_i(t) + q_{i-1}(t)\Delta t - q_i(t)\Delta t. \quad (2.2)$$

## 2.2. Route choice

In this section, we consider a traffic network with nine intersections as presented in Figure 1. The network contains 12 origins and destinations. It is assumed that all the links, which are the road stretch between two signalized intersections, have the same length and each one consists of five cells in the longitudinal direction. We should point out that this work can be extended to asymmetric networks.

More than one route are available for the vehicles to travel from one origin to a destination. In order to obtain the set of suitable routes for a given origin–destination (OD) pair, some constraints must be imposed. The length of each suitable route between one OD pair in terms of number of cells needs to be constrained so that unnecessary detouring can be eliminated.

$$l_m^{ij} < L^{ij} + l_{max}, \quad (2.3)$$

where  $l_m^{ij}$  denotes the length of the route numbered  $m$  between the OD of  $i$  and  $j$  subject to  $i, j = 1, 2, \dots, 12$ ,  $L^{ij}$  is the minimum distance of the OD pair, and  $l_{max}$  is a predetermined coefficient.  $L^{ij}$  can be obtained with the popular shortest path algorithm, also known as Dijkstra's algorithm (Dijkstra, 1959).

Let  $r^{ij}$  denote the set of suitable routes from origin  $i$  to destination  $j$ . For a chosen suitable route, say route  $m$ , between the OD pair, we define a proportion  $\Psi_m^{ij}$  of the traffic flow following the route as,

$$\Psi_m^{ij} = \frac{\exp(\mu(V_n^{ij} + \beta \ln P_n^{ij}))}{\sum_{n \in r^{ij}} \exp(\mu(V_n^{ij} + \beta \ln P_n^{ij}))}, \quad (2.4)$$

where  $V_n^{ij}$  denotes the utility of route  $n$ , which consists of components such as travel time, distance, and fuel consumption. In this paper, the utility is represented by the corresponding route distance, and  $V$  equals the reciprocal of the length of the considered route.  $\mu$  and  $\beta$  are constant coefficients. Specifically,  $\mu$  is a scale parameter assumed equal for all routes in the choice set (Bliemer & Bovy, 2008). The path-size factor for route  $m$   $P_m^{ij}$  is given as,

$$P_m^{ij} = \sum_{a \in S_m^{ij}} \left( \frac{\rho_a}{S_m^{ij}} \right) \frac{1}{\sum_{n \in r^{ij}} \delta_{a,n}^{ij} \left( \frac{S_n^{ij}}{S_m^{ij}} \right)^\gamma}, \quad (2.5)$$

where  $S_m^{ij}$  is the set of links constituting route  $m$ ,  $\rho_a$  denotes the length of link  $a$ ,  $\bar{S}^{ij}$  represents the shortest route from  $i$  to  $j$ ,  $\gamma$  is a scaling parameter, and  $\delta_{a,n}^{ij}$  is the indicator that is equal to 1 if link  $a$  is on route  $n$  and 0 otherwise. This is known as the path-size logit (PSL) model (Ben-Akiva & Bierlaire, 1999; Bliemer & Bovy, 2008). Compared with other logit models such as the multinomial logit model (MNL) where the route choice probabilities for a given route choice set

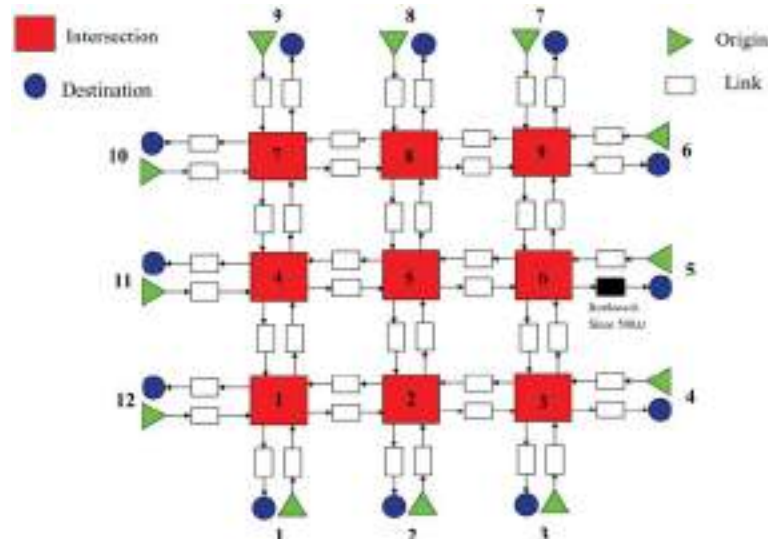


Figure 1. The traffic network configuration with nine intersections.

are usually determined directly from the route utilities and the traffic network structure is often neglected, the PSL model takes into account the overlap in route alternatives by introducing the path-size factor  $P$  and is able to better model the route choice behaviors in network. The reader can also refer to these references for the constants used in the model.

The route choices between origins and destinations, and the proportions of the traffic flow taking each route have been incorporated in the CTM. Every proportion of the traffic flow assigned to a route when entering the network is assumed to follow through the route.

It should be noted that the route choice model is used in this study to introduce multiple possible routes for each origin–destination pair, while only one route is considered in most related researches. In this way, the complex urban traffic network can be more accurately modeled, and the effectiveness of the proposed optimization method can be better demonstrated in the realistic traffic scenarios. For simplicity, only the fixed route fraction based on the physical distance is considered in this study, and more advanced route choice models can be easily adopted in the proposed method. In addition, the route choice behavior of travelers in response to the variation of travel time is not considered in this paper, and the interaction between demand and supply is supposed to be formulated if that is incorporated in the traffic model.

### 2.3. Intersection model

Figure 2 shows the cell representation of links and intersection. In each link, the traffic flow is separated into two parts in opposite moving directions and

further divided into left turn, through and right-turn flows based on traveling routes. The exclusive cells are used for each traffic flow in the CTM. For instance, when a vehicle arrives at a new link after crossing the intersection, it is assumed to be in the left-turn cells if it prefers to turn left at the approaching intersection.

In the intersection area, center cells  $L_i$ ,  $T_i$  and  $R_i$  ( $i = 1, 2, 3, 4$ ) are used to model vehicle conflict and spillback behaviors (Lertworawanich, Kuwahara, & Miska, 2011; Aziz & Ukkusuri, 2011, 2012; Pohlmann and Friedrich, 2010; Beard & Ziliaskopoulos, 2006; Li & Sun, 2016b). The merging and diverging behaviors at the intersection are complicated since different vehicle traveling routes are considered. Each sending cell ( $L_i$ ,  $T_i$  and  $R_i$ ) releases traffic flow to three downstream receiving cells across the intersection, and the receiving cell receives the traffic flow from three upstream sending cells. Daganzo (1995) modeled the merging and diverging movements at junction for the basic CTM. A cell can be in either merging or diverging regime. However, the cells in the intersection center are a part of both merging and diverging regimes. In the following, additional rules for the merging and diverging models are presented to handle different route choices.

Figure 3 shows the merging and diverging scheme at intersection. Take the situation in Approach 3 for example. The traffic flows coming to Approach 3 from other approaches can merge into cell  $m$ ,  $n$ , and  $p$ , depending on their traveling routes, i.e., whether they need to turn left, go straight, and turn right at the next intersection. For each of the other three approaches, three virtual cells are used to classify the flow directions in advance. Take Approach 1 for instance. The left-turn flow in cell  $i$  can be classified into three categories. One is to turn left at the next

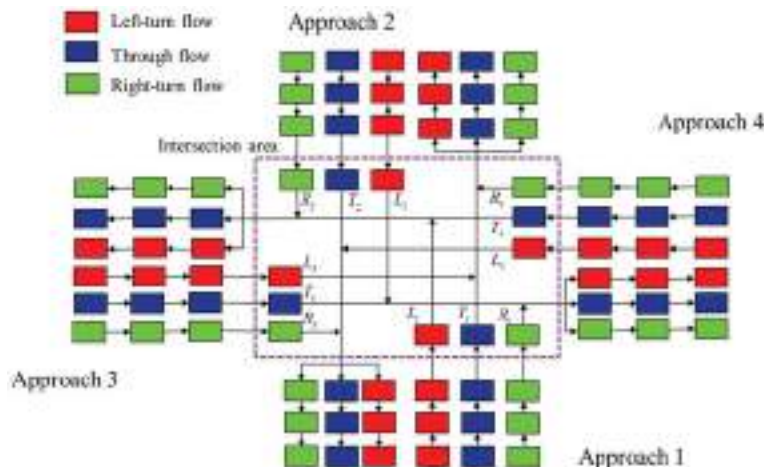
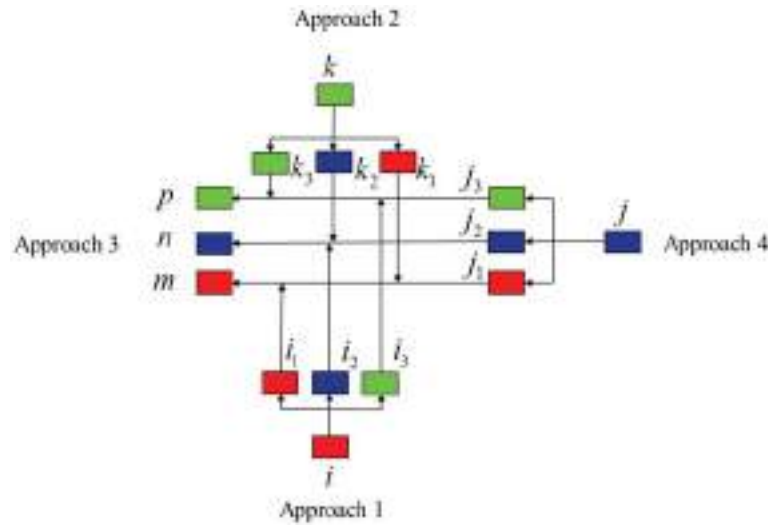


Figure 2. Cell representation of link and intersection in CTM.



**Figure 3.** Merging and diverging scheme at intersection.

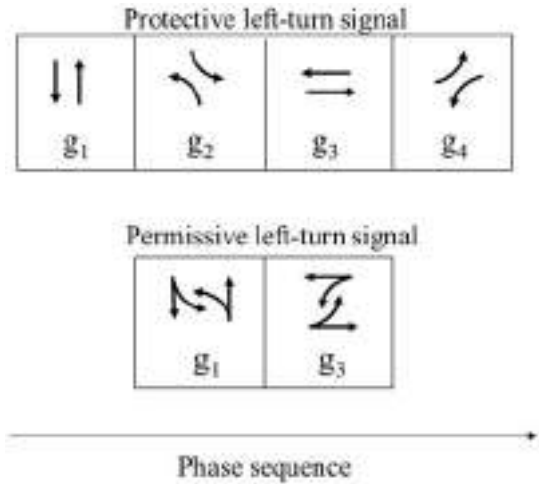
intersection after crossing this intersection, one is to go through at the next intersection, and the other is to turn right at the next intersection. Virtual cells  $i_1$ ,  $i_2$ , and  $i_3$  receive the flows, respectively. The flows in cell  $i_1$ ,  $j_1$ , and  $k_1$  that prefer to turn left at the next intersection after crossing this intersection merge into cell  $m$ . The situations in cell  $n$  (go straight at the next intersection) and  $p$  (turn right at the next intersection) are similar with those in cell  $m$ . Therefore, we have modeled the crossing behavior as separate merging and diverging processes (Daganzo, 1995). It should be pointed out that the virtual cells in the intersection do not exist in space, and they are used for modeling logic.

To simulate the effect of signal control, the outflow capacity of the signal control cells at the intersection (see Figure 2) can be formulated as a binary variable that switches between null and saturation flow  $Q_{\max}$  (Lo, 1999, 2001),

$$Q_i(t) = \begin{cases} Q_{\max}, & t \in \text{green phase} \\ 0, & t \in \text{red phase} \end{cases} \quad (2.6)$$

where cell  $i$  is the signal control cell and the signal phase for each traffic flow is introduced in the following.

Two kinds of left-turn signals are considered in this study, i.e., protective and permissive left-turn signal. When the protective left-turn signal is used, the left-turning traffic is protected by stopping the opposing through traffic. When the permissive left-turn signal is used, the left-turning traffic is permitted to cross through the opposing flow. The two signal control schemes are presented in Figure 4. The following deceleration rules are introduced to deal with the conflicts between left-turn and opposing through traffic with permissive left-turn signal (Li & Sun, 2016a).



**Figure 4.** Protective and permissive left-turn signal patterns at intersection.

Take the conflicts between the through traffic in Approach 1 and left-turn traffic in Approach 2 for instance. An upper bound for the mixed traffic flow holds,

$$q_{L_2}(t) + q_{T_1}(t) \leq c_{left} Q_{T_1}, \quad (2.7)$$

where  $c_{left}$  is a predefined conflict coefficient for mixed traffic flows. The two traffic flows follow the merging rules (Daganzo, 1995) in this situation.

The traffic network simulation model formulated above is based on the popular CTM (Daganzo, 1994, 1995) that is consistent with the kinematic wave theory (Lighthill & Whitham, 1955). Additional rules are added to the original framework to model route choices between OD pairs and vehicle turning and conflicting behaviors at intersections. When the diversity of the route choices is not considered, the presented model becomes the basic CTM. Additionally,

it is assumed in this paper that vehicles would follow the through, left-turn and right-turn lanes immediately when they enter the intersection. However in reality, exclusive turn lanes are usually provided when approaching the intersection and that can be further incorporated to improve the traffic model.

### 3. Model predictive controllers

#### 3.1. Control design

Let  $N_{scale}$  denote the number of intersections in a grid network. The geometry configuration with  $N_{scale}=9$  is presented in Figure 1. Two kinds of network models with different complexities are considered in this study, i.e., a simple and advanced network. In this section, two corresponding controls are formulated.

##### 3.1.1. Simple network

In the simple network, only straight-through vehicles are considered. Vehicle turning and different route choices are neglected. For instance, in Figure 2, the  $L_i$  and  $R_i$  cells in the intersection area ( $i = 1, 2, 3, 4$ ) and the corresponding left-turn and right-turn link cells in the CTM are ignored. This simplified model is able to reflect the basic traffic characteristics and has been widely used for network signal coordination investigations (Lertworawanich et al., 2011).

In most studies in the current literature, some key parameters such as the signal cycle length are fixed in the control design. More information can be included in the proposed method.

In the simple network, two parameters need to be optimized for each intersection. They are the green time  $g_1$  for the *vertical* traffic in one cycle and the corresponding cycle time  $t_{cycle}$ . Hence,  $t_{cycle}-g_1$  represents the green time for the *horizontal* traffic in the cycle. Therefore, the control sequence  $\mathbf{k}(t_{ctrl})$  at a certain control step  $t_{ctrl}$  can be formulated as,

$$\mathbf{k}(t_{ctrl}) = [\mathbf{C}_1(t_{ctrl}), \mathbf{C}_2(t_{ctrl}), \dots, \mathbf{C}_p(t_{ctrl})], \quad (3.1)$$

where

$$\begin{aligned} \mathbf{C}_i(t_{ctrl}) &= [\mathbf{S}_1(c_i|t_{ctrl}), \mathbf{S}_2(c_i|t_{ctrl}), \dots, \mathbf{S}_{N_{scale}}(c_i|t_{ctrl})], \\ \mathbf{S}_j(c_i|t_{ctrl}) &= [g_1^j(c_i|t_{ctrl}), t_{cycle}^j(c_i|t_{ctrl})], \\ i &= 1, 2, \dots, p, j = 1, 2, \dots, N_{scale}. \end{aligned}$$

$\mathbf{C}_i(t_{ctrl})$  denotes the general network signal plan for the  $i^{th}$  cycle since control step  $t_{ctrl}$  and  $\mathbf{S}_j(c_i|t_{ctrl})$  represent the signal plan of the  $j^{th}$  intersection in the  $i^{th}$  cycle.  $g_1^j(c_i|t_{ctrl})$  and  $t_{cycle}^j(c_i|t_{ctrl})$  denote the vertical traffic green time and cycle length of intersection  $j$  in

the  $i^{th}$  cycle, respectively.  $c_i$  represents the  $i^{th}$  cycle in the signal plan. To decrease the online computational complexity,  $p$  is introduced to represent the number of planned cycles, such that

$$\mathbf{C}_i(t_{ctrl}) = \mathbf{C}_p(t_{ctrl}), i > p, \quad (3.2)$$

that indicates the signal plan for the  $p^{th}$  cycle will be used repeatedly in the following cycles until the control changes.

Some constraints on the parameters apply,

$$\begin{aligned} 2t_{pha,min} &\leq t_{cycle} \leq t_{cyc,max}, \\ t_{pha,min} &\leq g_1 \leq t_{cycle} - t_{pha,min}, \end{aligned} \quad (3.3)$$

where  $t_{pha,min}$  is the minimum time of each phase and  $t_{cyc,max}$  denotes the maximum cycle time.

#### 3.2. Advanced network

The advanced network model considers vehicle turning and different route choices between origins and destinations, which makes it more complicated and realistic than the simple network. The advanced model is studied to show the extensibility of the proposed optimization framework.

Since vehicle turning is involved, two left-turn signal types are included in the control design, i.e., protective and permissive one. As Figure 4 shows, when protective left-turn signal is used, left-turning traffic is protected by stopping the opposing through traffic. When permissive left-turn signal is used, left-turning traffic is permitted to cross through the opposing flow. Detailed information about the effect of left-turn signal type on intersection performance can be found in Li and Sun (2016a).

Let  $S_{per}$  denote whether permissive left-turn signal is used ( $S_{per}=1$ ) or protective left-turn signal is used ( $S_{per}=0$ ).  $g_1, g_2, g_3,$  and  $g_4$  are the corresponding green time in each phase as presented in Figure 4. The control sequence can be formulated as,

$$\mathbf{k}(t_{ctrl}) = [\mathbf{C}_1(t_{ctrl}), \mathbf{C}_2(t_{ctrl}), \dots, \mathbf{C}_p(t_{ctrl})], \quad (3.4)$$

where

$$\begin{aligned} \mathbf{C}_i(t_{ctrl}) &= [\mathbf{S}_1(c_i|t_{ctrl}), \mathbf{S}_2(c_i|t_{ctrl}), \dots, \mathbf{S}_{N_{scale}}(c_i|t_{ctrl})], \\ \mathbf{S}_j(c_i|t_{ctrl}) &= [S_{per}^j(c_i|t_{ctrl}), g_1^j(c_i|t_{ctrl}), g_2^j(c_i|t_{ctrl}), g_3^j(c_i|t_{ctrl}), t_{cycle}^j(c_i|t_{ctrl})], \\ i &= 1, 2, \dots, p, j = 1, 2, \dots, N_{scale}. \end{aligned}$$

$S_{per}^j(c_i|t_{ctrl})$  denotes the left-turn signal type of intersection  $j$  in the  $i^{th}$  cycle since control step  $t_{ctrl}$ .  $g_2^j(c_i|t_{ctrl})$  and  $g_3^j(c_i|t_{ctrl})$  represent the corresponding green time in the two phases as Figure 4 shows.

Some parameter constraints are imposed. When  $S_{per}^j(c_i|t_{ctrl}) = 1,$

$$\begin{aligned} g_2^j(c_i|t_{ctrl}) &= 0, g_4^j(c_i|t_{ctrl}) = 0, \\ g_3^j(c_i|t_{ctrl}) &= t_{cycle}^j(c_i|t_{ctrl}) - g_1^j(c_i|t_{ctrl}), \end{aligned} \quad (3.5)$$

and when  $S_{per}^j(c_i|t_{ctrl}) = 0$ ,

$$g_4^j(c_i|t_{ctrl}) = t_{cycle}^j(c_i|t_{ctrl}) - g_1^j(c_i|t_{ctrl}) - g_2^j(c_i|t_{ctrl}) - g_3^j(c_i|t_{ctrl}). \quad (3.6)$$

In addition,

$$\begin{aligned} t_{pha,min} &\leq t_{cycle}^j(c_i|t_{ctrl}) \leq t_{cyc,max}, \\ t_{pha,min} &\leq g_k^j(c_i|t_{ctrl}) \leq t_{cycle} - t_{pha,min}, \\ &k = 1, 2, 3, 4. \end{aligned} \quad (3.7)$$

### 3.2. Control implementation

The proposed algorithm repeatedly solves the optimization problem online in a rolling horizon to derive the control for the next control horizon. The time domain updating scheme of the control is presented in Figure 5. The prediction horizon  $N_p$  and control horizon  $N_c$  are assumed to be constant. The model described in Section 2 is used for traffic prediction of the simple and advanced networks.

Since the feasibility of the dynamic control is very important, the online computing time for the control design is investigated. In the time step  $t_{ctrl,p} = t_{ctrl} - t_{delay}$  where  $t_{delay}$  is the delayed time step, the information of the current network is assumed to be collected by the control and used for prediction, which includes the number of vehicles in each cell, traffic demands, *etc.* The online control design is assumed to be completed in  $t_{delay}$ . The prediction horizon is the following  $N_p$  steps from  $t_{ctrl,p}$ .

The optimized control  $\mathbf{k}(t_{ctrl})$  can be derived by control step  $t_{ctrl}$  where  $t_{ctrl} = n_{int} \times N_c$  and  $n_{int}$  is an integer. The multi-objective optimization formulation described in Section 4 is used in the control design, and the GA is employed to find the Pareto optimal solutions through simulations over the prediction horizon.

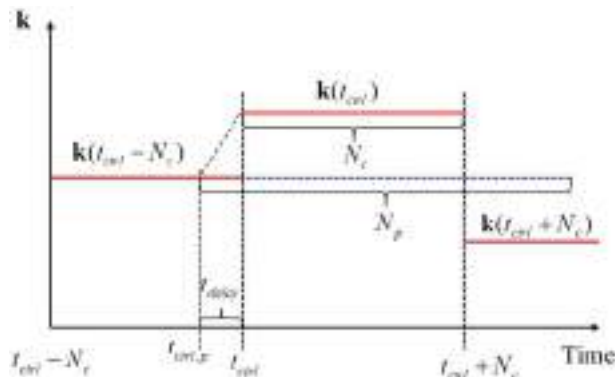


Figure 5. Timing of dynamic signal predictive control scheme.

At the next control step, the prediction horizon is shifted one step forward, and the optimization is carried out again with the updated network information. The control system benefits from the rolling horizon framework with feedbacks from the real traffic periodically, which makes the proposed control robust to uncertainties and disturbances.

In the implementation of the proposed control method, the system needs to be stable in the transition process between two controls. When the next control horizon starts, the new signal cycle should not be implemented until the current cycle expires.

All online optimization algorithms rely on efficient computational tools to be successful. In this study, the CPU parallel computing is employed to accelerate the control design in real time. With the help of CTM, GA, and parallel computing, the optimal controls can be obtained efficiently online. This study will be reported in Section 5.

## 4. Network multi-objective optimization

In most of the existing signal multi-objective optimization methods, multiple objectives are combined into a single one using weights. In this way, the MOP becomes single objective optimization (SOP) problem. In this section, a traffic network MOP is formulated. A novel multi-objective optimization algorithm on dynamic signal control is proposed where multiple objectives are optimized simultaneously.

### 4.1. Multi-Objective optimization problem

A MOP can be stated as (Li & Sun, 2018b),

$$\min_{\mathbf{k} \in Q} \{ \mathbf{F}(\mathbf{k}) \} = \min_{\mathbf{k} \in Q} [o_1(\mathbf{k}), o_2(\mathbf{k}), \dots, o_{n_{obj}}(\mathbf{k})], \quad (4.1)$$

where  $\mathbf{F}: U \rightarrow \mathbf{R}^{n_{obj}}$  is the map that consists of the objective functions  $o_i: U \rightarrow \mathbf{R}^1$ .  $n_{obj}$  is the number of objective functions.  $\mathbf{k} \in U$  is a  $c$ -dimensional vector of design parameters.  $U \subset \mathbf{R}^c$  is a bounded domain and can be generally expressed by inequality and equality constraints,

$$U = \{ \mathbf{k} \in \mathbf{R}^c | \text{subject to constraints on } \mathbf{k} \}. \quad (4.2)$$

The concept of *dominancy* (Pareto, 1971) defined below plays an important role in defining the optimal solution of the MOP.

#### Definition 1

- Let  $\mathbf{u}, \mathbf{z} \in \mathbf{R}^n$ . The vector  $\mathbf{u}$  is said to be less than  $\mathbf{z}$  ( $\mathbf{u} <_p \mathbf{z}$ ) if  $u_i < z_i$  for all  $1 \leq i \leq n$ . The relation  $\leq_p$  is defined analogously.



2. A design vector  $\mathbf{u} \in U$  is called *dominated* by another vector  $\mathbf{z} \in U$  ( $\mathbf{z} \prec \mathbf{u}$ ) with respect to MOP (4.1) if  $\mathbf{F}(\mathbf{z}) \leq_p \mathbf{F}(\mathbf{u})$  and  $\mathbf{F}(\mathbf{u}) \neq \mathbf{F}(\mathbf{z})$ . Otherwise,  $\mathbf{u}$  is called non-dominated by  $\mathbf{z}$ .

Therefore, a design parameter vector  $\mathbf{u}$  can be considered *better* than  $\mathbf{z}$  with respect to MOP (4.1) if  $\mathbf{u}$  dominates  $\mathbf{z}$ . The definition of optimality or the *best* solution of the MOP is now straightforward.

### Definition 2

1. A point  $\mathbf{u} \in U$  is called Pareto optimal or a Pareto point of MOP (4.1) if there is no  $\mathbf{z} \in U$  that dominates it.
2. The set of all Pareto optimal solutions is called the *Pareto set*, denoted as  $P_{set}$ .

$$P_{set} := \{\mathbf{u} \in U : \mathbf{u} \text{ is a Pareto point of MOP (4.1)}\}. \quad (4.3)$$

3. The image  $\mathbf{F}(P_{set})$  of  $P_{set}$  is called the *Pareto front*. The Pareto set and Pareto front typically form an  $(n_{obj}-1)$ -dimensional manifold under certain mild assumptions on the MOP (Hillermeier, 2001; Li & Sun, 2018a). Recent studies seem to suggest that the Pareto front may have fine structures for MOPs of complex dynamical systems (Hernández et al., 2013). A survey of the methods for the solution of MOPs can be found in Jones, Mirrazavi, and Tamiz (2002) and Marler and Arora (2004).

In general, five measures of network performance can be used as control objectives (Li & Sun, 2014),

1. maximizing system throughputs.
2. minimizing traveling delays.
3. maximizing intersection crossing volume.
4. avoiding spillbacks.
5. enhancing traffic safety.

Considering the terminal cells in the network as the boundary, system cumulative outflow can be obtained by adding all the outflows of the terminal cells at each step.

$$f_{out,m} = \sum_{i \in T_{out,cell}} \sum_t q_i(t), \quad (4.4)$$

where  $T_{out,cell}$  denotes the terminal outflow cell set in the network and  $f_{out,m}$  is the system cumulative outflow.

To fully take advantage of the network capacity, maximizing network inflow can be a control objective, which is obtained as,

$$f_{in,m} = \sum_{i \in T_{in,cell}} \sum_t q_i(t), \quad (4.5)$$

where  $T_{in,cell}$  denotes the terminal inflow cell set and  $f_{in,m}$  is the cumulative system inflow.

Vehicle traveling delay in each cell at time  $t$  can be estimated by subtracting the outflow of the cell from its occupancy (Lo, 2001). Once the delay has been determined at the cell level, it can be easily calculated at the system level by adding the delays of the involved cells.

$$d_{all} = \sum_i \sum_t (n_i(t) - q_i(t) \Delta t), \quad (4.6)$$

where  $d_{all}$  denotes the cumulative vehicle delay in the whole network.

Vehicle crossing volume at intersections can be calculated by summing up the outflow of the intersection cells from different approaches.

$$c_{j,m} = \sum_{i \in I_{j,cell}} \sum_t q_i(t), \quad (4.7)$$

$$c_{total,m} = \sum_{j=1}^{N_{scale}} c_{j,m}, \quad (4.8)$$

where  $c_{j,m}$  denotes the cumulative crossing volume at intersection  $j$ ,  $I_{j,cell}$  is the cell set of the intersection, and  $c_{total,m}$  represents the cumulative crossing volume of all the intersections.

Spillback is a situation when the vehicles on the approach of the downstream intersection overflow backward to the subject intersection. That leads to an intersection blockage, occasionally followed by a gridlock. Take approaches 1 and 2 for instance (Figure 2), a spillback is defined as (Lertworawanich et al., 2011),

$$\begin{aligned} n_{T_3}(t) > 0 \text{ or } n_{T_4}(t) > 0 \text{ or} \\ n_{L_3}(t) > 0 \text{ or } n_{L_4}(t) > 0, \end{aligned} \quad (4.9)$$

when the green phase (left-turn or through) in approaches 1 and 2 is active.

In the advanced network model, when the permissive left-turn signal is used, the left-turn and through flows share the same green phase that may lead to conflicts (Li & Sun, 2016a). In this study, traffic safety is measured by the potential conflicts between the two traffic flows. With respect to approaches 1 and 2 (see Figure 2), an emergency situation for traffic safety is defined as,

$$\begin{aligned} n_{L_1}(t) + n_{T_2}(t) > n_{emerg,cri} \text{ or} \\ n_{L_2}(t) + n_{T_1}(t) > n_{emerg,cri}, \end{aligned} \quad (4.10)$$

where  $n_{emerg,cri}$  is a predetermined coefficient. At each time step in the simulation, if the above equation is satisfied, the situation will be considered not safe.

Therefore, the network MOP can be formulated as,

$$\begin{aligned} \max_{\mathbf{k} \in Q} [f_{out,m}, f_{in,m}, c_{j,m}, c_{total,m}], \\ \min_{\mathbf{k} \in Q} [d_{all}, n_{emerg}, n_{spill}]. \end{aligned} \quad (4.11)$$

$n_{emerg}$  and  $n_{spill}$  denote the cumulative number of left-turn emergencies (Equation (4.10)) and spillbacks (Equation (4.9)), respectively. That indicates  $n_{emerg}$  will increase by 1 at each simulated time step if Equation (4.10) is satisfied, and similarly,  $n_{spill}$  will also increase by 1 at each time step if Equation (4.9) is satisfied. To simplify the MOP statement, we assume all the objectives are to be minimized. The network MOP becomes,

$$\min_{\mathbf{k} \in Q} \{ \mathbf{F}(\mathbf{k}) \}, \quad (4.12)$$

where

$$\mathbf{F}(\mathbf{k}) = [f_{out}, f_{in}, d_{all}, c_j, c_{total}, n_{emerg}, n_{spill}], \quad (4.13)$$

for advanced network optimization,

$$\mathbf{F}(\mathbf{k}) = [f_{out}, f_{in}, d_{all}, c_j, c_{total}, n_{spill}], \quad (4.14)$$

for simple network optimization, and

$$\begin{aligned} f_{out} &= f_{out,max} - f_{out,m}, \\ f_{in} &= f_{in,max} - f_{in,m}, \\ c_j &= c_{max} - c_{j,m}, \\ c_{total} &= N_{scale} \times c_{max} - c_{total,m}, \\ j &= 1, 2, \dots, N_{scale}. \end{aligned} \quad (4.15)$$

$f_{out,max} > 0, f_{in,max} > 0, c_{max} > 0$ , and  $c_{total,max} > 0$  are large coefficients to make the corresponding  $f_{out}$ ,  $f_{in}$ ,  $c_j$  and  $c_{total}$  positive for better presentation of the optimization results. In addition, proper constraints of the objective evaluations are needed to keep the traffic performance within an acceptable range.

$$\begin{aligned} \max f_{out} &\leq f_{out,lim}, \\ \max f_{in} &\leq f_{in,lim}, \\ \max c_j &\leq c_{lim}, \\ \max c_{total} &\leq c_{total,lim}, \\ \max d_{all} &\leq d_{all,lim}, \\ \max n_{emerg} &\leq n_{emerg,lim}, \\ \max n_{spill} &\leq n_{spill,lim}, \end{aligned} \quad (4.16)$$

where  $f_{out,lim}, f_{in,lim}, c_{lim}, c_{total,lim}, d_{all,lim}, n_{emerg,lim}$ , and  $n_{spill,lim}$  are the corresponding limits. The parameter bounds are set in order to help the convergence of the optimization, since the traffic control performance in applications must be in a reasonable range. Moreover, the bounds also restrict the design space that can accelerate the optimization computing in real time.

## 4.2. Genetic algorithm

The GA is a heuristic search method belonging to the family of evolutionary algorithms (Greenhalgh & Marshall, 2000; Deb, Pratap, Agarwal, & Meyarivan 2002), and the nondominated sorting genetic algorithm (NSGA) is adopted in this study. Crossover and

mutation are the two main genetic operators for searching new solutions starting from the current population. A single-point crossover technique is adopted in this study to generate new and improved solutions. A unique mutation method is used, i.e., the value presented in the particular cell is replaced by another randomly generated number. This improves the search space explored by the algorithm. The crossover and the mutation operator are performed with a given crossover and mutation probability.

Before the crossover and mutation operations, individuals are selected from the current generation to be parents. The individuals are selected based on the fitness, a value that reflects the quality of an individual. The larger the fitness value of an individual, the higher its chance of survival and reproduction. The superior ones are selected to be stochastically modified to create a new population for the next generation. For a MOP, a ranking operation is implemented to determine the fitness. The rank of an individual is generally determined by its Pareto dominance in the present population.

A Pareto-set filtering procedure is introduced to record the so-called Pareto-front solutions that are Pareto optimal among all solutions ever encountered by the algorithm. At each generation, after the ranking of the current generation, all nondominated individuals in the current generation are copied and put into an independent Pareto set. When new solutions are added into the set, a new round of dominance check is performed. Therefore, dominated ones are discarded and real nondominated points are reserved. With this repetitive procedure, the population eventually converges to the optimal solution.

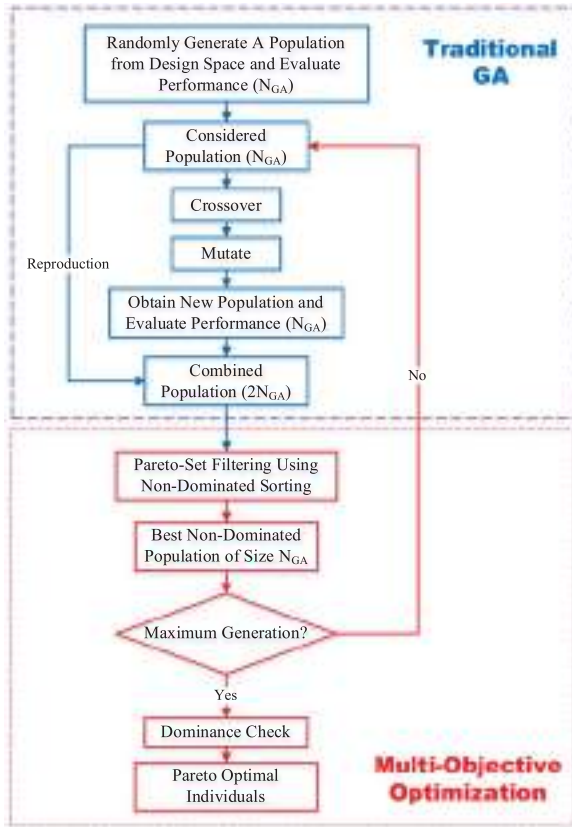
The flow chart of the multi-objective optimization GA is presented in Figure 6. For the detailed information of the algorithm, the readers are referred to Deb et al. (2002).

## 4.3. Control selection

After the MOP is solved with GA, multiple layers of Pareto fronts can be obtained in the final generation (Deb et al., 2002). The Pareto set corresponding to the best Pareto front is used as the solution set in this study. To facilitate the user to pick up a control from the set to implement, we propose an algorithm that operates on the Pareto front.

Let  $f_{i,min}$  denote the minimum of the  $i^{th}$  objective in the Pareto front and  $f_{i,max}$  be the corresponding maximum. Define a vector,

$$\mathbf{F}_{ideal} = [f_{1,min}, f_{2,min}, \dots, f_{n_{obj},min}]. \quad (4.17)$$



**Figure 6.** Flowchart of the multi-objective optimization genetic algorithm.  $N_{GA}$  denotes the size of the corresponding population.

where  $F_{ideal}$  is considered to be an ideal point in the objective space. Let  $n_s$  denote the number of Pareto solutions. To eliminate the effect of different scale of the objectives, the Pareto front is normalized as,

$$\bar{f}_{i,j} = \frac{f_{i,j} - f_{i,\min}}{f_{i,\max} - f_{i,\min}}, \quad (4.18)$$

$$i = 1, 2, \dots, n_{obj},$$

$$j = 1, 2, \dots, n_s,$$

where  $f_{i,j}$  is the  $i^{th}$  objective of the  $j^{th}$  optimal solution, and  $\bar{f}_{i,j}$  represents the normalized value of  $f_{i,j}$ . The normalized objective vector  $\bar{\mathbf{F}}_i$  reads,

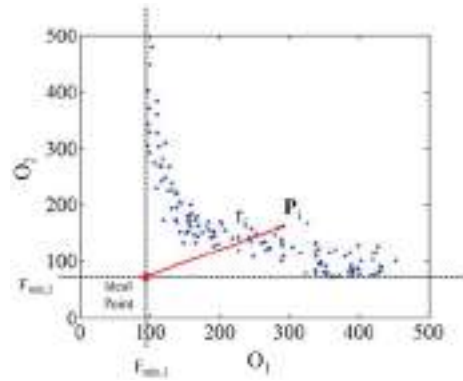
$$\bar{\mathbf{F}}_i = [\bar{f}_{1,i}, \bar{f}_{2,i}, \dots, \bar{f}_{n_{obj},i}], \quad (4.19)$$

and its norm is denoted as  $r_i = \|\bar{\mathbf{F}}_i\|$ . Let  $r_{\max}$  and  $r_{\min}$  denote the maximum and minimum of the norm.

Let  $n_{per}$  denote the percentage of control designs in the Pareto set such that their corresponding radii in the Pareto front are among the  $n_{per}$  percent smallest ones. We refer these controls as the top  $n_{per}$  percent. A special set of top controls consists of the so-called *knee points* (Hernández et al., 2013). They are defined as,

$$\mathbf{k}_{knee} = \left\{ \mathbf{k}_i \mid i = \min_{1 \leq i \leq n_s} r_i \right\}. \quad (4.20)$$

**Figure 7** illustrates the concept of control selection with two objectives. It should be noted that there are



**Figure 7.** Illustration of the control selection from a 2-dimensional Pareto front.

**Table 1.** The parameters used in this study.

Parameter	Value	Parameter	Value	Parameter	Value
$\omega$	13.9m/s	$v_{free}$	13.9m/s	$c_{total,max}$	90,000veh
$c_{left}$	.7	$Q_{max}$	2.5veh/ $\Delta t$	$c_{total,lim}$	45,000veh
$l$	75m	$t_{pha,min}$	2 $\Delta t$	$c_{max}$	10,000veh
$k_{jam}$	0.12veh/m	$f_{max}$	10,000veh	$c_{lim}$	5,000veh
$n_{emerg,cri}$	4	$f_{traffic,lim}$	9,000veh	$f_{out,max}$	50,000veh
$l_{max}$	2	$n_{emerg,lim}$	2,000	$f_{out,lim}$	30,000veh
$t_{delay}$	$\Delta t$	$d_{all,lim}$	30,000veh $\Delta t$	$f_{in,max}$	50,000veh
$t_{cyc,max}$	20 $\Delta t$	$n_{spill,lim}$	200	$f_{in,lim}$	30,000veh

different ways to normalize the Pareto front. Furthermore, the normalization affects the classification of the controls. The normalization used in this paper is intuitive and simple to implement.

## 5. Optimization results

We present a case study with  $N_{scale} = 9$  intersections in a grid network with simple model as shown in **Figure 1**. Simulation time step  $\Delta t$  is 5 seconds and the whole optimization duration was 1000 $\Delta t$ . The popular CTM coefficients in Lo (2001) are used in this study. The initial network is assumed to be empty. The default values of the key parameters in the proposed method are  $p = 2$ ,  $N_c = 20$ , and  $N_p = 30$ . In the GA, the default population size is 1,000 and the number of generations is 30. The influence of  $N_p$  and  $N_c$  on the optimization results and the selections of the coefficients in GA are discussed in **Section 5.1**. The other parameters are presented in **Table 1**.

The evaluations of objective functions in GA are independent from each other. Hence, the CPU parallel computing can be used to accelerate the control design process. The computations of all examples reported in this paper are conducted on a PC with 2.50GHz CPU and 12 cores. The programming is in C++. Each scenario of the online optimizations is simulated for 20 times. The mean values of the

**Table 2.** Varying traffic demands in simple network optimization with nine intersections. The duration of each time period is 500s. Unit: veh/h.

Demand from origin	Time period									
	1	2	3	4	5	6	7	8	9	10
1	1,600	1,550	1,300	1,650	1,600	1,800	2,000	1,600	1,650	1,450
2	1,400	1,200	1,500	1,800	2,000	1,850	1,750	1,400	1,400	1,400
3	1,700	1,500	1,800	2,200	2,200	2,100	2,000	1,900	1,800	1,700
4	1,600	1,800	1,400	1,400	1,400	1,200	1,000	800	800	1,000
5	1,500	1,400	1,300	1,600	1,800	1,800	1,900	2,000	1,500	1,500
6	1,500	1,600	1,400	1,400	1,400	1,500	1,700	1,800	1,900	1,600
7	1,700	1,600	1,600	1,500	1,700	1,800	1,700	1,600	1,400	1,600
8	1,400	1,200	1,200	1,300	1,600	1,600	1,800	1,800	2,000	1,800
9	1,600	1,600	1,700	1,800	1,500	1,500	1,500	1,300	1,200	1,200
10	1,600	1,800	2,000	2,200	2,200	2,100	2,000	1,600	2,000	1,600
11	1,700	1,500	1,400	1,300	1,500	1,600	1,800	1,400	1,300	1,500
12	1,800	1,800	1,800	1,700	1,600	1,500	1,400	1,800	1,800	1,600

**Table 3.** The dynamic optimization results with different control demands in simple network with nine intersections. The standard deviations share the same scales with the corresponding mean values.

control		optimization results							
		$\bar{d}_{all}/1000(veh\Delta t)$	$\sigma_{d_{all}}$	$\bar{c}_1/100(veh)$	$\sigma_{c_1}$	$\bar{c}_8/100(veh)$	$\sigma_{c_8}$	$\bar{c}_{total}/100(veh)$	$\sigma_{c_{total}}$
Knee Point	$\mathbf{k}_{knee}$	455.1(-3.5%)	5.92	38.0(+1.2%)	.40	38.2(-8.8%)	.23	353.1(-5.5%)	1.31
Top 20%	$\mathbf{k}_{20}$	461.6	11.4	38.8	.42	38.6	.25	353.5	2.55
Top 50%	$\mathbf{k}_{50}$	452.2	16.3	38.5	.70	38.5	.39	357.4	4.62
Extreme Designs	$\mathbf{k}_1$	430.1(-9.0%)	7.04	39.7(-13.0%)	.61	40.0(+2.7%)	.35	353.0(-9.7%)	1.35
	$\mathbf{k}_2$	489.2(-21.7%)	6.21	36.9(-6.2%)	.22	39.5(-6.1%)	.28	358.1(-5.9%)	1.51
	$\mathbf{k}_3$	479.2(-25.3%)	5.57	38.9(-7.2%)	.41	37.8(-5.0%)	.26	352.3(-9.6%)	1.42
	$\mathbf{k}_4$	472.0(-19.6%)	9.42	38.4(-4.7%)	.27	38.2(-11.8%)	.16	349.1(-3.9%)	2.36
Time-invariant Controls	$\mathbf{k}_{f1}$	614.2	8.32	49.1	.31	46.5	.24	478.6	2.07
	$\mathbf{k}_{f2}$	578.1	9.18	48.2	.26	45.6	.28	457.5	1.68

statistics are taken as the result of optimization. The corresponding standard deviations are provided.

It is often to consider the predetermined traffic demands in the control design (Lo, 2001; Lertworawanich et al., 2011). The varying traffic demands considered in this case are shown in Table 2. The MOP formulation is,

$$\min_{\mathbf{k} \in U} [d_{all}, c_1, c_8, c_{total}], \quad (5.1)$$

where  $[d_{all}, c_1, c_8, c_{total}]$  is a subset of the MOP formulation in Equation (4.14), and the rest of the objectives will be analyzed in Section 5.3.

The post-processing algorithm proposed in Section 4.3 is applied to the Pareto fronts obtained in each control step. The user can pick a control from the top  $n_{per}$  subset of the Pareto optimal designs according to the performance balance over different objectives. The performances of the knee point  $\mathbf{k}_{knee}$  and two other controls  $\mathbf{k}_{20}$  and  $\mathbf{k}_{50}$  are shown in Table 3.  $\mathbf{k}_{20}$  is a control randomly picked from the top  $n_{per} = 20\%$  subset of the Pareto set, and  $\mathbf{k}_{50}$  is randomly picked from the top  $n_{per} = 50\%$ .

As we have seen in Hernández et al. (2013), the performances of all the Pareto optimal controls are usually bounded by those of the extreme designs with one of the objectives being the global minimum. In fact, the extreme designs are the results of single objective optimization problems. All other Pareto

optimal controls represent various compromises of these extreme designs. Let  $\mathbf{k}_1, \mathbf{k}_2, \mathbf{k}_3$ , and  $\mathbf{k}_4$  denote the extreme designs with the best performance of  $d_{all}$ ,  $c_1$ ,  $c_8$ , and  $c_{total}$ , respectively. Their performances are shown in Table 3. Two controls,  $\mathbf{k}_{f1}$  and  $\mathbf{k}_{f2}$ , that are time-invariant and therefore nonoptimal, are listed in the table for comparison. For  $i = 1, 2, \dots, N_{scale}$ ,  $\mathbf{k}_{f1}$  represents the control with  $g_1^i = 6$  and  $t_{cycle}^i = 12$ , and  $\mathbf{k}_{f2}$  represents that with  $g_1^i = 10$  and  $t_{cycle}^i = 20$ .

Furthermore, in order to evaluate the effectiveness of the dynamic control, the static optimization is carried out with the varying traffic demand. Assume that the traffic flow fluctuations in the whole simulation period are available in advance. The Pareto set over the entire simulation period can be obtained with the proposed method, and the static optimization can be thus considered as an offline approach. The numbers in the brackets in Table 3 represent the improvement (negative value) and deterioration (positive value) in the performance by the dynamic control compared with that by the static optimization responding to the same traffic demand.

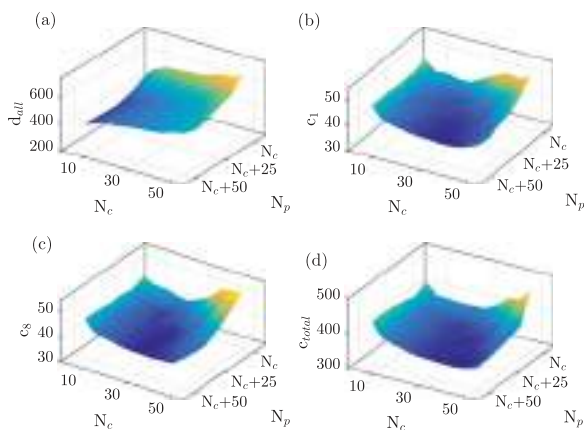
It can be observed that the proposed controls can achieve the optimal and balanced traffic performance with respect to the objectives. While  $\mathbf{k}_{knee}$ ,  $\mathbf{k}_{20}$ , and  $\mathbf{k}_{50}$  share similar balanced mean values of the performance objectives, their standard deviations increase

individually, representing different compromises. For example, when minimizing the vehicle overall delay has the highest priority, the extreme control  $\mathbf{k}_1$  delivers the performance objective  $d_{all}=437.4$ , smaller than that with other controls. Similar characteristics can be observed with respect to  $\mathbf{k}_2, \mathbf{k}_3$ , and  $\mathbf{k}_4$ . Significant improvements in the network performance are obtained with the predictive control compared with the two time-invariant controls. The proposed online optimization algorithm is able to achieve the optimal traffic performance in different situations.

In addition, the proposed dynamic optimization method outperforms the static optimization in most cases. Better performance can be achieved by the online predictive control with the varying traffic demand. The average CPU computing time for the control design in this scenario is 1.69s and the standard deviation is 0.11s. That is an acceptable delay since  $t_{delay} = 1\Delta t$  (5s) is reserved for online computation.

### 5.1. Notes on the algorithm

In this section, the influence of the prediction and control horizon on the optimization performance is investigated first. Figure 8 shows the optimization results of  $d_{all}$ ,  $c_1$ ,  $c_8$ , and  $c_{total}$  with different combinations of  $N_p$  and  $N_c$ . The population size is 1000, and the number of generations is 30 in GA. Specialized control demand is used to optimize each objective. For instance,  $\mathbf{k}_1$  is used in sub-figure (a) for a better view of the performance of  $d_{all}$ . Correspondingly,



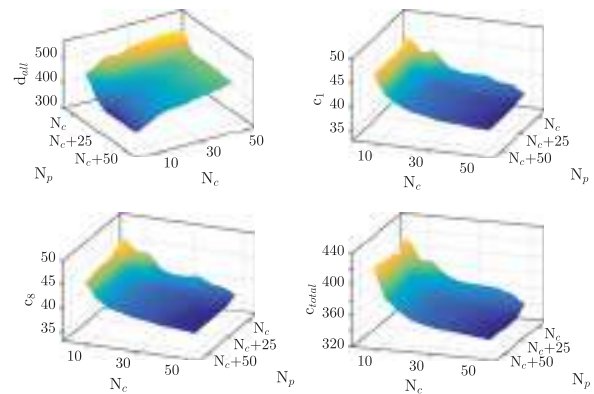
**Figure 8.** The optimization performance  $d_{all}$  (veh $\Delta t$ /1000),  $c_1$  (veh/100),  $c_8$  (veh/100), and  $c_{total}$  (veh/100) with different combinations of  $N_c$  ( $\Delta t$ ) and  $N_p$  ( $\Delta t$ ). The population size is 500, and the number of generations is 30 in GA. The brighter color indicates the higher level, while the darker color means the lower level. Subfigures (a), (b), (c), and (d) show the optimization performance of  $d_{all}$ ,  $c_1$ ,  $c_8$ , and  $c_{total}$  with control demand  $\Phi_1, \Phi_2, \Phi_3$ , and  $\Phi_4$ , respectively.

$\mathbf{k}_2, \mathbf{k}_3$ , and  $\mathbf{k}_4$  are used in sub-figures (b), (c), and (d), respectively.

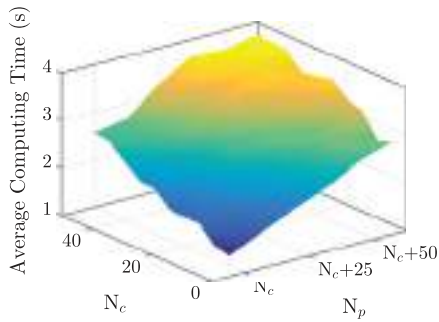
In general, larger  $N_p$  leads to better optimization results with identical  $N_c$ . However, two display patterns of the influence of  $N_c$  are observed with the four sub-figures. For network delay  $d_{all}$  in sub-figure (a), shorter control horizon provides better optimization. For  $c_1$ ,  $c_8$ , and  $c_{total}$  in the other sub-figures which fall in the performance category of intersection crossing volume, the optimal results can be obtained with the medium  $N_c$ . Similarly, the optimization results with the knee point  $\mathbf{k}_{knee}$  are presented in Figure 9. While the influence of  $N_c$  on  $d_{all}$  resembles that in Figure 8, larger  $N_c$  yields better optimization performance of intersection crossing volume in the range of interest.

Therefore, while longer prediction horizon is more beneficial for the system generally, the effect of control horizon on optimization has underlying relationships with objective selection and control demand. On one hand, shorter control horizon leads to better optimization results of reducing vehicle traveling time in network. On the other hand, the intersection crossing volume usually increases with relatively longer  $N_c$ . No general influence of control horizon is observed with respect to different performance objectives.

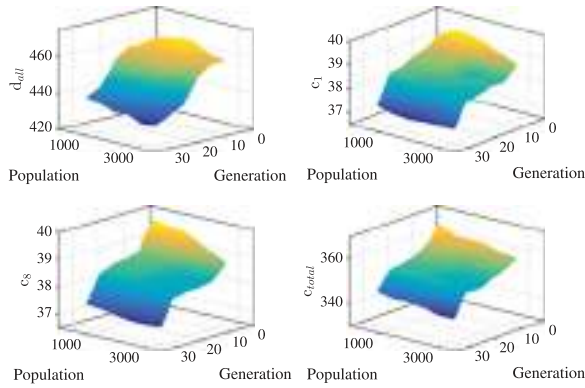
The corresponding average CPU computing time in the controller design process with different  $N_c$  and  $N_p$  is presented in Figure 10. In general, longer prediction horizon results in larger online computational burden. Considering the trade-off between optimization performance and online computing time,  $N_c = 20$  and  $N_p = 30$  are used in this paper.



**Figure 9.** The optimization performance  $d_{all}$  (veh $\Delta t$ /1000),  $c_1$  (veh/100),  $c_8$  (veh/100), and  $c_{total}$  (veh/100) with different combinations of  $N_c$  ( $\Delta t$ ) and  $N_p$  ( $\Delta t$ ). The most balanced controller over different objectives  $\Phi_i$  is used. The population size is 500, and the number of generations is 30 in GA. The brighter color indicates the higher level while the darker color means the lower level.



**Figure 10.** The average CPU computing time in the control design process with different combinations of control and prediction horizons. The brighter color indicates the higher level while the darker color means the lower level.

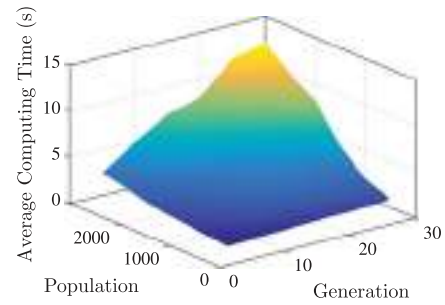


**Figure 11.** The optimization performance  $d_{all}$  ( $veh\Delta t/1000$ ),  $c_1$  ( $veh/100$ ),  $c_8$  ( $veh/100$ ), and  $c_{total}$  ( $veh/100$ ) with different combinations of population size and generations in GA. The brighter color indicates the higher level, while the darker color means the lower level.

Furthermore, guidelines are presented on the selection of the coefficients in GA in the proposed control framework.

Figure 11 shows the optimization performance of the four objectives with different combinations of population size and generations in GA. It can be observed that in general, more generations lead to better optimization performance significantly with identical population. However, the improvements of larger population size on the optimization results are limited. The corresponding average CPU computing time in the controller design process with different population size and generations is presented in Figure 12. It is clear that larger population size and more generations result in larger online computational burden.

Based on the performance characteristics of the parameters in GA, small population size (1,000) and many generations (30) are preferred in this study to achieve good optimization results without much online computing time.



**Figure 12.** The average CPU computing time in the control design process with different combinations of population size and generations in GA. The brighter color indicates the higher level, while the darker color means the lower level.

It should be pointed out that while the controller formulation can be high dimensional (see Equations (3.1) and (3.4)), many constraints hold (see Equations (3.3), (3.5), (3.6), and (3.7)), and some parameters ( $S_{per}$ ) are not continuous in the design space. Therefore, the feasible design space is limited, and the population size of 1000 in GA is able to cover the space and achieve the optimal results.

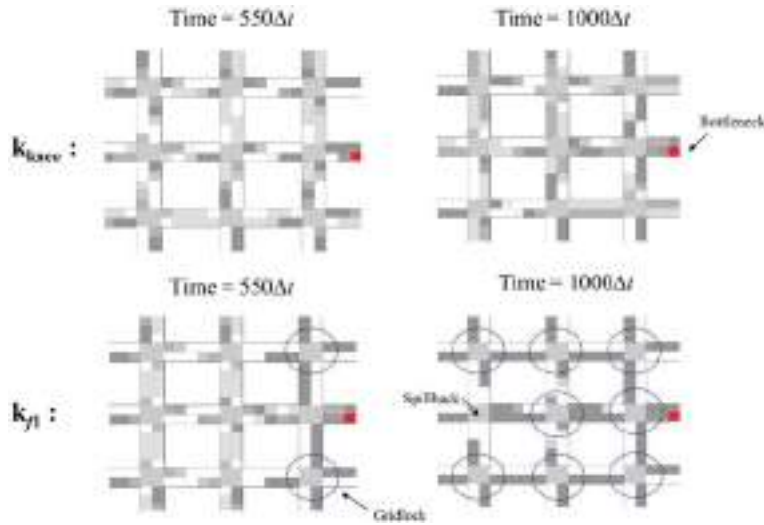
In addition, when the network scale becomes larger ( $N_{scale}$ ), more design parameters are introduced (see Equations (3.1) and (3.4)). However, due to the parameter constraints in the GA (see Equations (3.3), (3.5), (3.6), and (3.7)), the number of actual function evaluations in the controller design process with identical GA coefficients tends to decrease with larger network scale. The optimization performance is not at the same level (deteriorates in most cases) when  $N_{scale}$  increases with fixed population size and the number of generations. Therefore, the influence of the network scale on the computational burden can be hardly investigated with identical optimum.

In summary, the trade-off between online computing time and optimization performance has to be made when the proposed method is used in practice.

## 5.2. Network disturbance

In this section, as Figure 1 shows, a road bottleneck is created at time step  $500\Delta t$  to investigate the network performance against disturbance with the proposed dynamic optimization method. The outflow capacity  $Q$  of one cell in the bottleneck link becomes  $.5$  since  $500\Delta t$  due to traffic accident.

In order to capture the network congestion in real time, screenshots at time steps  $550\Delta t$  and  $1000\Delta t$  in the simulation animations are presented in Figure 13. The upper panel shows the network situations with the proposed online predictive control. The control  $k_{knee}$  is used. The lower panel shows the congested



**Figure 13.** Network screenshots in disturbance with controls  $\mathbf{k}_{knee}$  and  $\mathbf{k}_{f1}$  at time steps 550 and 1000 $\Delta t$ .

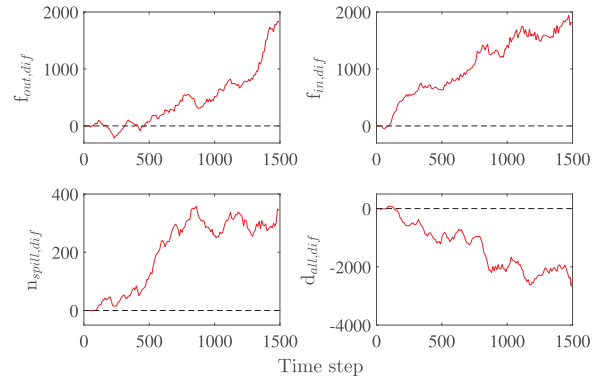
network circumstance with the time-invariant control  $\mathbf{k}_{f1}$ . The darker color in the links indicates more vehicles are in the cell. It can be observed that the traffic keeps flowing and no significant congestion occurs under the dynamic control, while the network jam can hardly dissipate and gridlock forms under the time-invariant control. The proposed dynamic control reacts fast to the network capacity changes with disturbance and achieves optimal performance.

### 5.3. Objective selection

Different from the long-term network optimization, short-term predictive control is more focused on instant benefits in the prediction horizon even though cumulative performance is considered in the optimization. The advantages lie in that the control is able to switch online according to the traffic demand. For instance, the control can be promptly tuned if the traffic engineer in charge of the network operation prefers to relieve more congestion at an intersection at some time rather than focusing on reducing overall vehicle delay. However, the drawback is that the combination of local optimum in each control step does not lead to global optimum for some performance indices. In this section, the influence of different objective selections on network optimization performance is studied considering more objectives as,

$$\min_{\mathbf{k} \in U} [f_{out}, f_{in}, n_{spill}, d_{all}, c_{total}]. \quad (5.2)$$

Let us define five new extreme controls  $\mathbf{k}_{1,new}$ ,  $\mathbf{k}_{2,new}$ ,  $\mathbf{k}_{3,new}$ ,  $\mathbf{k}_{4,new}$ , and  $\mathbf{k}_{5,new}$  that correspond to the best performance of  $f_{out}$ ,  $f_{in}$ ,  $n_{spill}$ ,  $d_{all}$ , and  $c_{total}$  respectively. In the following, two levels of traffic demands are applied to the network. Figure 14 shows



**Figure 14.** The influence of objective selection on network cumulative performance with high-traffic demands.

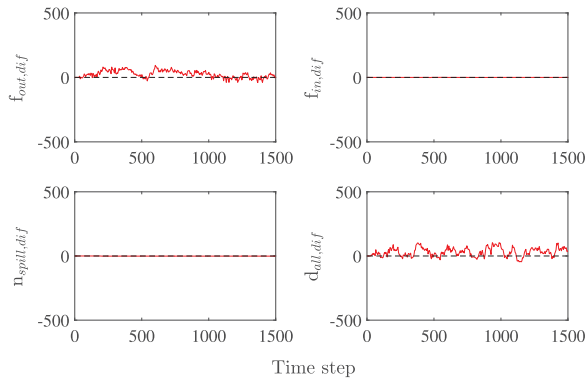
the influence of different control demands on the network cumulative performance  $f_{out}$ ,  $f_{in}$ ,  $n_{spill}$  and  $d_{all}$  with high traffic volume as listed in Table 2. Define a variable,

$$f_{out,dif}(t) = f_{out}^1(t) - f_{out}^5(t), \quad (5.3)$$

where  $f_{out}^1(t)$  denotes the cumulative system output  $f_{out}$  at time step  $t$  with control  $\mathbf{k}_{1,new}$  and  $f_{out}^5(t)$  is that with control  $\mathbf{k}_{5,new}$ .  $f_{out,dif}(t)$  reflects the cumulative improvements (negative) or deterioration (positive) in  $f_{out}$  by the dynamic control which uniquely focuses on improving  $f_{out}$  in each control horizon, compared with that by a “typical” control. Without loss of generality,  $\mathbf{k}_{5,new}$  is taken as the typical control for comparison. Similarly,

$$\begin{aligned} f_{in,dif}(t) &= f_{in}^2(t) - f_{in}^5(t), \\ n_{spill,dif}(t) &= n_{spill}^3(t) - n_{spill}^5(t), \\ d_{all,dif}(t) &= d_{all}^4(t) - d_{all}^5(t). \end{aligned} \quad (5.4)$$

It can be observed in the traffic congestions that the indices  $f_{out,dif}$ ,  $f_{in,dif}$ , and  $n_{spill,dif}$  remain positive in



**Figure 15.** The influence of objective selection on network cumulative performance with low-traffic demands.

the steady state. This indicates pursuing local optimum of some performance such as system output  $f_{out}$ , inflow  $f_{in}$ , and spillbacks  $n_{spill}$  in each control period does not lead to global optimum in the long term with high traffic demands. As a comparison, the difference in the cumulative vehicle delay of control  $\mathbf{k}_{4,new}$  and  $\mathbf{k}_{5,new}$  is presented in the lower-right sub-figure.  $d_{all,dif}$  is negative in the steady state, suggesting a reduction of overall delays at both local and global optimum.

Figure 15 shows the influence of different controls with low traffic volume. The traffic demands are 30 percent of those in Table 2. It can be observed that  $f_{in,dif}$  and  $n_{spill,dif}$  almost remain zero, which indicates no significant difference in the network inflow with different controls. The spillback hardly occurs with

low traffic volume. Small fluctuations around zero are observed in  $f_{out,dif}$  and  $d_{all,dif}$ . Different from congested situations, optimization results are less affected by the selection of objectives with low traffic volume.

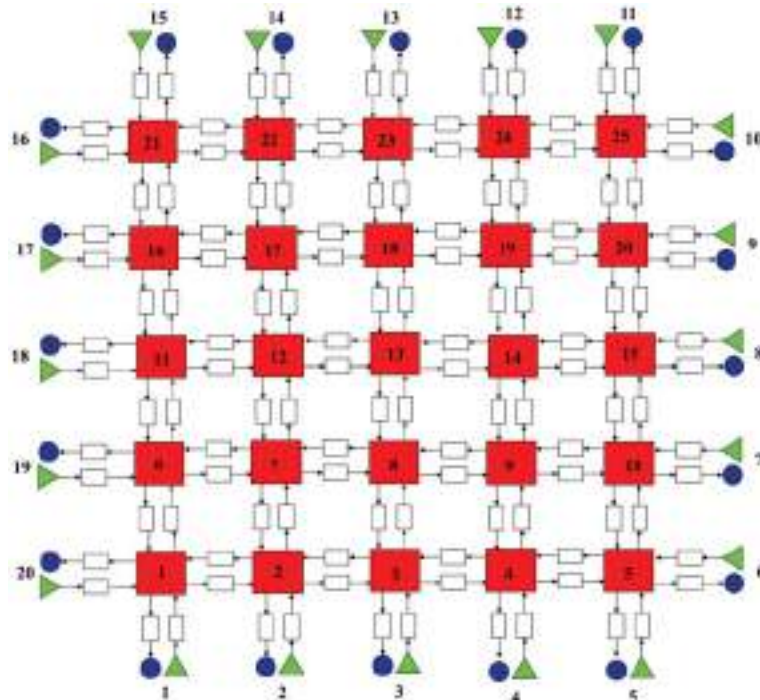
In summary, the objectives for optimization in the predictive control framework should have both short-term and long-term benefits. With high-traffic demands, some indices such as  $f_{out}$ ,  $f_{in}$ , and  $n_{spill}$  are not suitable to be objectives, while the influence of objective selections on the results of optimization is weak with low-traffic volume.

#### 5.4. Extensibility to larger scale

The proposed dynamic multi-objective optimization method can handle large-scale network with acceptable computing time. In this section, the traffic network shown in Figure 16 with  $N_{scale} = 25$  is investigated to show the extensibility. The constant traffic demands are considered for this example and are shown in Table 4. The dynamic control scheme and parameters are similar with those in Section 5. The MOP for this example is formulated as,

$$\min_{\mathbf{k} \in U} [d_{all}, c_1, c_{23}, c_{total}]. \quad (5.5)$$

Table 5 gives the optimization results. The controls are similar to those in Table 3. It can be observed that the proposed control is able to effectively optimize the large-scale network. The average computing time of



**Figure 16.** The traffic network configuration with 25 intersections.



**Table 4.** Time-invariant traffic demands in simple network optimization with twenty-five intersections. Unit: veh/h.

Origin	Traffic demands												
	1	2	3	4	5	6	7	8	9	10	11	12	13
Demand	1,600	1,800	1,500	1,800	2,000	2,000	1,400	1,500	1,800	1,800	1,600	1,500	1,600
Origin	14	15	16	17	18	19	20	21	22	23	24	25	
Demand	1,800	1,600	1,600	2,000	2,200	1,600	1,600	1,700	1,800	1,800	1,400	1,800	

**Table 5.** The dynamic optimization results with different control demands in simple network with twenty-five intersections. The standard deviations share the same scales with the corresponding mean values.

Control		Optimization results							
		$\bar{d}_{all}/1000$ (veh $\Delta t$ )	$\sigma_{d_{all}}$	$\bar{c}_1/100$ (veh)	$\sigma_{c_1}$	$\bar{c}_{23}/100$ (veh)	$\sigma_{c_{23}}$	$\bar{c}_{total}/100$ (veh)	$\sigma_{c_{total}}$
Knee Point	$\mathbf{k}_{knee}$	1571 (−8.5%)	18.86	39.8 (+3.7%)	.21	39.1 (+2.7%)	.12	902.1 (−6.9%)	3.55
Top 20%	$\mathbf{k}_{20}$	1595	29.53	39.4	.31	38.8	.42	898.4	9.55
Top 50%	$\mathbf{k}_{50}$	1562	45.11	41.2	.86	38.1	.56	906.3	10.14
Extreme Designs	$\mathbf{k}_1$	1505 (−4.9%)	24.51	39.5 (−1.5%)	.17	40.2 (−0.4%)	.12	938.4 (+2.8%)	3.10
	$\mathbf{k}_2$	1640 (−21.8%)	20.87	38.4 (−4.6%)	.11	38.9 (−4.4%)	.26	892.3 (−8.2%)	2.11
	$\mathbf{k}_3$	1598 (−12.5%)	22.03	38.9 (−4.0%)	.21	37.3 (−6.8%)	.12	900.1 (−4.3%)	2.41
	$\mathbf{k}_4$	1591 (−5.5%)	18.44	38.8 (−6.0%)	.13	37.9 (−4.7%)	.26	881.2 (−4.8%)	4.43
Time-invariant Controls	$\mathbf{k}_{f1}$	1769	24.41	42.6	.24	42.1	.18	956.5	4.02
	$\mathbf{k}_{f2}$	1786	21.34	44.1	.36	39.6	.16	970.1	3.86

the control selection in this example is 1.53s, and the standard deviation is .09s. As discussed in Section 5.1, the online computing time decreases as the network scale becomes larger with the same parameters in GA since more constraints are imposed (see Equations (3.3), (3.5), (3.6), and (3.7)). The population size and number of generations are supposed to increase with larger network if the same quality of the optimization results as that for a different network scale is demanded.

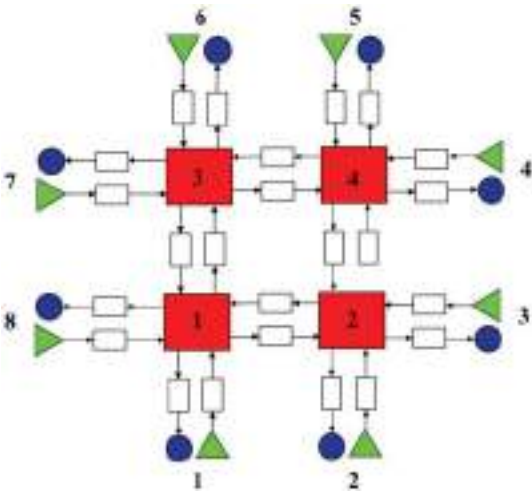
### 5.5. Extensibility to advanced model

The advanced network model is used to model complex traffic networks including vehicle turning and route choices. In this section, the proposed algorithm

is applied to an advanced network with  $N_{scale}=4$  as shown in Figure 17. Vehicle turning and different route choices between origins and destinations are considered. The origin–destination (OD) traffic demands are shown in Table 6. In each OD pair, the probabilities of different traveling routes are determined by the path-size logit model 2.2.  $t_{cyc,max}$  is  $40\Delta t$  in this case. The optimization scheme and parameters are similar with those in Section 5. The MOP for this network is formulated as,

$$\min_{\mathbf{k} \in U} [d_{all}, c_1, c_{total}, n_{emerg}]. \quad (5.6)$$

The optimization results are listed in Table 7. The controls are similar to those in Table 3. For  $i = 1, 2, \dots, N_{scale}$ , the control  $\mathbf{k}_{f1}$  with  $g_1^i = 6, g_2^i = 6, g_3^i = 6, t_{cycle}^i = 24$  and  $S_{per}^i = 0$  focuses on the safety, and  $\mathbf{k}_{f2}$  with  $g_1^i = 10, g_2^i = 0, g_3^i = 10, t_{cycle}^i = 20$  and  $S_{per}^i = 1$  aims at high traffic efficiency. It is observed that the proposed method is able to effectively handle online optimization with increased model complexity. The average computing time for the control design is 3.91s and the standard deviation is .40s. While more

**Figure 17.** The traffic network configuration with four intersections.**Table 6.** Time-invariant traffic demands between different origins and destinations in advanced network optimization with four intersections. Unit: veh/h.

Origin	Destination							
	1	2	3	4	5	6	7	8
1	N/A	100	200	50	100	400	200	100
2	100	N/A	200	30	450	100	50	80
3	70	100	N/A	144	150	100	150	380
4	50	100	120	N/A	80	80	400	120
5	50	290	100	120	N/A	120	80	80
6	450	120	80	70	100	N/A	120	70
7	100	110	100	370	120	80	N/A	80
8	150	60	380	100	60	80	70	N/A

**Table 7.** The dynamic optimization results with different control demands in advanced network with four intersections. The standard deviations share the same scales with the corresponding mean values.

Control		Optimization results							
		$\bar{d}_{all}/1000$ (veh $\Delta$ t)	$\sigma_{d_{all}}$	$\bar{c}_1/100$ (veh)	$\sigma_{c_1}$	$\bar{c}_{total}/100$ (veh)	$\sigma_{c_{total}}$	$\bar{n}_{emerg}$	$\sigma_{n_{emerg}}$
Knee Point	$k_{knee}$	279.4 (+2.2%)	4.51	46.1 (-3.7%)	.24	184.5 (-8.0%)	1.30	15.0 (+0.6%)	.21
Top 20%	$k_{20}$	276.1	5.33	43.4	.50	182.4	1.59	13.1	2.42
Top 50%	$k_{50}$	274.3	5.98	46.2	.61	186.1	2.04	14.8	6.40
Extreme Designs	$k_1$	266.5 (-4.9%)	4.53	46.0 (-11.3%)	.28	188.1 (-7.9%)	1.31	26.0 (+4.0%)	0.19
	$k_2$	275.1 (-13.8%)	3.92	43.0 (-14.7%)	.13	192.4 (-5.0%)	.91	18.9 (-5.2%)	.35
	$k_3$	291.1 (-15.0%)	3.14	45.5 (-7.7%)	.19	181.4 (-9.0%)	1.41	12.7 (-7.4%)	.13
	$k_4$	293.5 (-3.8%)	3.69	47.9 (-6.0%)	.29	190.3 (-4.4%)	.67	5.0 (-0.8%)	.12
Time-invariant Controls	$k_{r1}$	353.4	4.31	52.4	.31	224.6	.87	5.7	.15
	$k_{r2}$	310.9	2.87	50.2	.24	203.4	.96	42.7	1.28

realistic modeling issues and control choices can be considered with the advanced network, the computational burden rises significantly compared with the simple model. With the help of CTM, GA, and parallel computing, the online computing time can be acceptable without deterioration of the control performance. In order to decrease the computing time, the parameters such as population size, the number of GA generations, control and prediction horizons, and planned signal cycles can be tuned. The trade-off between online computational load and model accuracy has to be made when the proposed method is used in practice.

## 6. Conclusions

In this paper, a novel dynamic multi-objective optimization method on network signal predictive control is proposed. The CTM is used for traffic prediction. Two kinds of network models with different complexities are studied by simulations. Multiple operational objectives are optimized simultaneously with the online predictive control algorithm. A wide variety of objectives are considered such as maximizing system throughput, minimizing vehicle delay, avoiding spill-backs. The MOP is solved with the GA. An algorithm is proposed to assist the user to select and implement the Pareto optimal designs.

It is observed that the optimal traffic performance can be achieved by the dynamic control in different situations. The proposed control is able to optimize networks with different model complexities and scales. Network disturbance can be well handled. The computational burden of the online dynamic control is acceptable with the help of parallel computing. The influence of the key parameters in the proposed method on traffic behavior is investigated, and the trade-off between optimization performance and computational complexity is studied. The results of this study suggest that the proposed dynamic multi-

objective optimization of the traffic network system offers a new and promising approach.

It should be noted that in this study, the boundary conditions of the traffic network are assumed to be available. For instance, the network inflows can be monitored by the system. The CTM is only used to generate the short-term state estimation in the traffic network system given the boundary conditions. In further research, external predictors such as time series model and neural network are supposed to be adopted if the predictions of the boundary conditions are considered.

In addition, it is demonstrated in the case studies in the paper that the proposed method is able to handle operation uncertainty in the network, such as network disturbance and the variation of traffic demands. Since the proposed algorithm is based on real-time MPC method, it holds the potential to achieve strong robustness against uncertainty, disturbance, and model mismatch, which can be further developed in following researches.

This study has used the GA extensively, which is a stochastic approach. Other deterministic methods for solving the MOP such as the cell mapping method can be applied. In addition, only the simulations are used in this study. Real traffic data should be used to obtain the network state when the proposed method is applied to real traffic networks. Based on the proposed optimization method, further researches such as urban network evaluation and traffic navigation can be carried out. Different optimization algorithms such as cell mapping and partial swarm optimization (PSO) are also encouraged to be tested in the framework for comparisons.

## Disclosure statement

No potential conflict of interest was reported by the author(s).

## Funding

The material in this paper is based on work supported by grants (N170503012, N170308028) from the Fundamental

Research Funds for the Central Universities, and grants (11172197, 11332008 and 11572215) from the National Natural Science Foundation of China.

## References

- Aboudolas, K., Papageorgiou, M., & Kosmatopoulos, E. (2009). Store-and-forward based methods for the signal control problem in large-scale congested urban road networks. *Transportation Research Part C: Emerging Technologies*, 17(2), 163–174.
- Aziz, H., & Ukkusuri, S. (2012). Unified framework for dynamic traffic assignment and signal control with cell transmission model." *Transportation Research Record: Journal of the Transportation Research Board*, 2311(1), 73–84.
- Aziz, H. M. A., & Ukkusuri, S. V. (2011). An analytical framework for vehicular traffic signal control integrated with dynamic traffic assignment using cell transmission model. *Proceedings of the 14th International IEEE Conference on Intelligent Transportation Systems*, 1355–1360. In *Intelligent Transportation Systems (ITSC)*.
- Beard, C., & Ziliaskopoulos, A. (2006). System optimal signal optimization formulation. *Transportation Research Record: Journal of the Transportation Research Board*, 1978(1), 102–112.
- Ben-Akiva, M., & Bierlaire, M. (1999). Discrete choice methods and their applications to short term travel decisions. In *Handbook of Transportation Science*, Vol. 23 of *International Series in Operations Research & Management Science* (pp. 5–33). New York: Springer.
- Bliemer, M., & Bovy, P. (2008). Impact of route choice set on route choice probabilities. *Transportation Research Record: Journal of the Transportation Research Board*, 2076(1), 10–19.
- Cantarella, G. E., de Luca, S., Di Pace, R., & Memoli, S. (2015). Network signal setting design: Meta-heuristic optimisation methods. *Transportation Research Part C: Emerging Technologies*, 55, 24–45.
- Chen, X., Xiong, C., He, X., Zhu, Z., & Zhang, L. (2016). Time-of-day vehicle mileage fees for congestion mitigation and revenue generation: A simulation-based optimization method and its real-world application. *Transportation Research Part C: Emerging Technologies*, 63, 71–95.
- Chow, A. H. F., & Lo, H. K. (2007). Sensitivity analysis of signal control with physical queuing: Delay derivatives and an application. *Transportation Research Part B: Methodological*, 41(4), 462–477.
- Daganzo, C. F. (1994). The cell transmission model: A dynamic representation of highway traffic consistent with the hydrodynamic theory. *Transportation Research Part B: Methodological*, 28(4), 269–287.
- Daganzo, C. F. (1995). The cell transmission model, part II: Network traffic. *Transportation Research Part B: Methodological*, 29(2), 79–93.
- Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2), 182–197.
- Diakaki, C., Papageorgiou, M., & Aboudolas, K. (2002). A multivariable regulator approach to traffic-responsive network-wide signal control. *Control Engineering Practice*, 10(2), 183–195.
- Dijkstra, E. W. (1959). A note on two problems in connexion with graphs. *Numerische Mathematik*, 1(1), 269–271.
- Dotoli, M., Fanti, M. P., & Meloni, C. (2006). A signal timing plan formulation for urban traffic control. *Control Engineering Practice*, 14(11), 1297–1311.
- Geroliminis, N., Haddad, J., & Ramezani, M. (2013). Optimal perimeter control for two urban regions with macroscopic fundamental diagrams: A model predictive approach. *IEEE Transactions on Intelligent Transportation Systems*, 14(1), 348–359.
- Ghanim, M. S., & Abu-Lebdeh, G. (2015). "Real-time dynamic transit signal priority optimization for coordinated traffic networks using genetic algorithms and artificial neural networks." *Journal of Intelligent Transportation Systems*, 19(4), 327–338.
- Greenhalgh, D., & Marshall, S. (2000). Convergence criteria for genetic algorithms. *SIAM Journal on Computing*, 30(1), 269–282.
- Hale, D. K., Antoniou, C., Park, B. B., Ma, J., Zhang, L., & Paz, A. (2017). Revisiting the application of simultaneous perturbation stochastic approximation towards signal timing optimization. *Journal of Intelligent Transportation Systems*, 22(5), 1–11.
- Hegy, A., De Schutter, B., & Hellendoorn, J. (2005). Optimal coordination of variable speed limits to suppress shock waves. *IEEE Transactions on Intelligent Transportation Systems*, 6(1), 102–112.
- Hernández, C., Naranjani, Y., Sardahi, Y., Liang, W., Schütze, O., & Sun, J.-Q. (2013). Simple cell mapping method for multi-objective optimal feedback control design. *International Journal of Dynamics and Control*, 1(3), 231–238.
- Hillermeier, C. (2001). *Nonlinear multiobjective optimization: A generalized homotopy approach*. New York: Springer Science & Business Media.
- Jones, D. F., Mirrazavi, S. K., & Tamiz, M. (2002). Multi-objective meta-heuristics: An overview of the current state-of-the-art. *European Journal of Operational Research*, 137(1), 1–9.
- Lertworawanich, P., Kuwahara, M., & Miska, M. (2011). A new multiobjective signal optimization for oversaturated networks. *IEEE Transactions on Intelligent Transportation Systems*, 12(4), 967–976.
- Li, X., & Sun, J.-Q. (2014). Effect of interactions between vehicles and pedestrians on fuel consumption and emissions. *Physica A: Statistical Mechanics and Its Applications*, 416, 661–675.
- Li, X., & Sun, J.-Q. (2015). Studies of vehicle lane-changing to avoid pedestrians with cellular automata. *Physica A: Statistical Mechanics and Its Applications*, 438, 251–271.
- Li, X., & Sun, J.-Q. (2016). Effects of turning and through lane sharing on traffic performance at intersections. *Physica A: Statistical Mechanics and Its Applications*, 444, 622–640.
- Li, X., & Sun, J.-Q. (2016). Effects of vehicle-pedestrian interaction and speed limit on traffic performance of intersections. *Physica A: Statistical Mechanics and Its Applications*, 460, 335–347.

- Li, X., & Sun, J.-Q. (2017). Studies of vehicle lane-changing dynamics and its effect on traffic efficiency, safety and environmental impact. *Physica A: Statistical Mechanics and Its Applications*, 467, 41–58.
- Li and Sun(2018a)] Li, X., & Sun, J.-Q. (2018a). “Signal Multiobjective Optimization for Urban Traffic Network.”. *IEEE Transactions on Intelligent Transportation Systems PP*, 99, 1–9.
- Li, X., & Sun, J.-Q. (2018). Turning-lane and signal optimization at intersections with multiple objectives. *Engineering Optimization*, 1–19.
- Lighthill, M. J., & Whitham, G. B. (1955). On kinematic waves. II. A theory of traffic flow on long crowded roads. *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 229(1178), 317–345.
- Lin, S., De Schutter, B., Yugeng, X., & Hellendoorn, H. (2011). Fast model predictive control for urban road networks via MILP. *IEEE Transactions on Intelligent Transportation Systems*, 12(3), 846–856.
- Lin, S., Zhou, Z., & Xi, Y. (2013). Model-based traffic congestion control in urban road networks: analysis. *Transportation Research Record: Journal of the Transportation Research Board*, 2390(1), 112–120.
- Liu, H., Han, K., Gayah, V. V., Friesz, T. L., & Yao, T. (2015). Data-driven linear decision rule approach for distributionally robust optimization of on-line signal control. *Transportation Research Part C: Emerging Technologies*, 59, 260–277.
- Lo, H. K. (1999). A novel traffic signal control formulation. *Transportation Research Part A: Policy and Practice*, 33(6), 433–448.
- Lo, H. K. (2001). A cell-based traffic control formulation: strategies and benefits of dynamic timing plans. *Transportation Science*, 35(2), 148–164.
- Lo, H. K., & Chow, A. H. F. (2004). Control strategies for oversaturated traffic. *Journal of Transportation Engineering*, 130(4), 466–478.
- Marler, R. T., & Arora, J. S. (2004). Survey of multi-objective optimization methods for engineering. *Structural and Multidisciplinary Optimization*, 26(6), 369–395.
- Olia, A., Abdelgawad, H., Abdulhai, B., & Razavi, S. N. (2017). Optimizing the number and locations of freeway roadside equipment units for travel time estimation in a connected vehicle environment. *Journal of Intelligent Transportation Systems*, 21(4), 296–309.
- Oliveira, L. B., & Camponogara, E. (2010). Multi-agent model predictive control of signaling split in urban traffic networks. *Transportation Research Part C: Emerging Technologies*, 18(1), 120–139.
- Ozan, C., Baskan, O., Haldenbilen, S., & Ceylan, H. (2015). A modified reinforcement learning algorithm for solving coordinated signalized networks. *Transportation Research Part C: Emerging Technologies*, 54, 40–55.
- Papatzikou, E., & Stathopoulos, A. (2015). An optimization method for sustainable traffic control in urban areas. *Transportation Research Part C: Emerging Technologies*, 55, 179–190.
- Pareto, V. 1971. *Manual of Political Economy*. London: The Macmillan Press.
- Pohlmann, T., & Friedrich, B. (2010). “Online control of signalized networks using the cell transmission model.”. *Proceedings of the 13th International IEEE Conference on Intelligent Transportation Systems*, 1–6. In *Intelligent Transportation Systems (ITSC)*.
- Prashanth, L. A., & Bhatnagar, S. (2011). Reinforcement learning with function approximation for traffic signal control. *IEEE Transactions on Intelligent Transportation Systems*, 12(2), 412–421.
- Richards, P. I. (1956). Shock waves on the highway. *Operations Research*, 4(1), 42–51.
- Robertson, D. I., & Bretherton, R. D. (1991). Optimizing networks of traffic signals in real time – the SCOOT method. *IEEE Transactions on Vehicular Technology*, 40(1), 11–15.
- Yang, I., & Jayakrishnan, R. (2015). Real-time network-wide traffic signal optimization considering long-term green ratios based on expected route flows. *Transportation Research Part C: Emerging Technologies*, 60, 241–257.
- Zhang, X., & Chang, G.-L. (2014). An optimization model for guiding pedestrian–vehicle mixed flows during an emergency evacuation. *Journal of Intelligent Transportation Systems*, 18(3), 273–285.