Multi-Commodity Traffic Signal Control and Routing With Connected Vehicles

Felipe de Souza⬡, Rodrigo Castelan Carlson⬡, *Member, IEEE*,
Eduardo Rauh Müller⬡, and Konstantinos Ampountolas⬡, *Member, IEEE*

Abstract— A real-time traffic management policy that integrates traffic signal control and multi-commodity routing of connected vehicles in networks with multiple destinations is developed. The proposed policy is based on a multi-commodity formulation of the store-and-forward model and assumes all vehicles are able to exchange information with the infrastructure. Vehicles share information about their current location and final destination. Based on this information, the strategy determines both optimized signal timings at every intersection and vehicle-specific routing information at every link of the network. The control actions, i.e., signal times and routing information, are updated at every cycle and delivered by a finite horizon optimal control problem cast into a rolling horizon framework. The underlying optimization problem is convex, and thus the method is suitable for real-time operation in large networks. The method is validated via a micro-simulation study in networks with up to twenty intersections and, in all simulations, outperforms a real-time traffic-responsive signal control strategy that is based on a single-commodity store-and-forward model. The scalable computation effort for increasing network sizes and prediction horizon confirms the computational efficiency of the method.

Index Terms— Traffic signal control, multi-commodity routing, connected vehicles, store-and-forward model, rolling horizon.

I. INTRODUCTION

WITH the evolving vehicle automation and communication technology, an increasing share of vehicles is able to exchange information via vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communication architectures. New and enhanced traffic management strategies such as ramp metering, platooning, traffic information and routing, and dynamic pricing for managed lanes may benefit of this

emerging scenario with connected vehicles [1] due to the access to data and information not available before.

Traditional traffic signal control strategies rely on rough assumptions of traffic routing, being usually captured indirectly by link specific turning rates. Several strategies attempted to close this gap by considering dynamic traffic assignment based on user equilibrium [2], [3]. In this scenario, the control strategy considers the individual choice behavior, but cannot act on it. The approach followed here, however, is to tackle the traffic signal control and routing problem simultaneously by benefiting from V2I communication to provide both signal times and routing in multi-destination networks with connected vehicles [4], [5].

The proposed traffic management strategy is based on the store-and-forward modelling approach previously used for traffic-responsive signal control in urban road networks [6]–[10]. More specifically, we extend the rolling horizon single-commodity traffic signal control strategy [9] with a multi-commodity formulation [11]. Vehicles at each link are identified as different commodities according to their intended destination. Then, signal times and routing information are computed in real-time. Signal times are provided via traffic lights while routing is delivered to the vehicles via V2I in the form of destination-specific turning rates for every road link in the network. This information is used to guide vehicles (drivers) by informing their next downstream destination (road link) once they arrive on a given link. Control decisions are taken at every cycle based on a rolling horizon control approach, allowing the possibility of re-routing vehicles and updating signal times in real-time in response to disturbances.

The control problem is formulated as a convex optimization program whereby solutions are obtained in polynomial time even for large-scale networks [12]. The proposed policy is especially suitable for a likely forthcoming scenario in which vehicles will be connected, but not necessarily fully automated, and able to exchange information via V2I communication.

In summary, the contributions to the state-of-the-art are:

- A novel mathematical formulation of the store-and-forward model for integrated multi-commodity traffic signal control and routing with connected vehicles;
- A convex optimization program to solve the formulated problem via rolling horizon in polynomial time by generic solvers and therefore suitable for real-time operation;
- A scheme to operationalize the control decisions into specific routing of individual vehicles, while the underlying formulation is based on a macroscopic model;

Felipe de Souza is with the Energy Systems, Argonne National Laboratory, Lemont, IL 60439 USA (e-mail: fdesouza@anl.gov).

Rodrigo Castelan Carlson and Eduardo Rauh Müller are with the Department of Automation and Systems, Federal University of Santa Catarina, Florianópolis 88040-900, Brazil (e-mail: rodrigo.carlson@ufsc.br; edurauh@gmail.com).

Konstantinos Ampountolas is with the Department of Mechanical Engineering, University of Thessaly, 38334 Volos, Greece (e-mail: k.ampountolas@uth.gr).

- An application of the proposed strategy via a micro-simulation study, which models features not captured by the underlying macroscopic control model.

Section II reviews related work. Section III develops the multi-commodity formulation for the traffic signal control and routing problem. It also discusses the scheme to operationalize routing. Section IV presents a micro-simulation case study of the proposed strategy for three grid networks of varying sizes and demand scenarios. Finally, Section V provides conclusions and directions for future work.

II. RELATED WORK

In early attempts to tackle the combined traffic signal control and routing problem from a control systems perspective (see, e.g., [13], [14]), connected vehicles were not considered. This is mainly attributed to the limited vehicular communication capabilities and roadside infrastructure to allow real-time data exchange via V2V and V2I architectures. To overcome this problem, [13], [14] pursued a decentralized approach. The travel times along a link were assumed fixed. A control law with the objective of reducing total delay was then derived for a multi-destination network based on the concept of cost-to-go from an origin link to a destination link by taking as input the queues at the links at the start of the traffic cycle. The signal times are calculated by an optimal control based on the same principles employed by [6] for a model of a single intersection. The delays induced by red lights in traffic signals are used as a component in the calculation of routing. The resulting strategy combining traffic signal control and routing was demonstrated to be capable of reducing link queues for a network with four intersections. The nature of the methods used for deriving the control laws, renders the method little demanding computationally.

Another decentralized approach was introduced by [15], in this case assuming connected vehicles and perfect communication. They departed from a single-commodity to a multi-commodity formulation in the context of back-pressure-based traffic control methods. The back-pressure single-commodity formulation for traffic signal control activates pre-timed phases, instead of affecting phase duration, according to the "pressure" exerted by queues at intersection approaches. In the multi-commodity formulation there is one commodity per destination, i.e., queues are known per link and per destination similar to what we propose in Section III. In this way, two new back-pressure algorithms were developed to handle both signal phase activation and routing in a decoupled manner. In other words, the solutions are combined instead of integrated into a single problem. Moreover, since the algorithms are decentralized, decisions are taken and applied locally with knowledge of adjacent queues only, thereby being rather efficient computationally. Routes are updated dynamically at every intersection and communicated to the vehicles. The back-pressure algorithms were evaluated in microsimulation for a realistic network with twenty-four intersections and constant demand. Vehicles were distributed more evenly across the network when compared to the single-commodity case, improving congestion, throughput and travel times. The method was extended by [16] to take into account, roughly, the time needed

for a vehicle to cross a road section. The microsimulation results for a grid network with sixteen intersections and time-varying demand has shown an improvement over the previous formulation. The scheme to operationalize aggregated turning rates into specific decisions for individual vehicles proposed in our study is conceptually similar to the one used in [16].

An iterative approach was proposed by [17]. Based on origin and destination (OD) information, current traffic signal plans and current travel times in the network, a routing problem is solved, followed by the solution of a traffic signal problem based on traffic flow distribution and queue lengths at the intersection. The resulting signal plan is used by the routing control in the next iteration. The design is modular, allowing different combinations of routing strategy and traffic signal control. The authors proposed a dynamic routing strategy based on varying and stochastic travel times in the solution of a shortest path problem. They also proposed two traffic signal control strategies, one of them a modified max-pressure algorithm. Thus, while the routing control is performed globally, the signal control is decentralized as in the previous approaches. All inputs are obtained from connected vehicles and infrastructure via V2I communication. The strategies were evaluated in microsimulation and by the use of a communication network simulator for the vehicular communication. The approach showed improvements in a 3×10 grid network. However, although the decentralized signal control is rather efficient, the routing optimization computational times increase as the cube of the number of intersections.

A centralized approach based on a macroscopic model with the goal of minimizing total time spent was proposed by [4]. The work is an extension of a previous Linear Quadratic Model Predictive Control [18] that was not able to preserve the correct OD relations, thus sending vehicles to the wrong destination. An enhanced macroscopic model and an heuristic are included to preserve OD relations rendering the method more suitable for multi-destination networks. The method was tested for a synthetic network modelled with a macroscopic model including two intersections, one expressway and seven OD pairs. The results showed an improvement in traffic conditions, close to system optimal, and that the approach was able to preserve OD relations. The computational times were suitable for real time operation, and it is expected that specifically designed algorithms may reduce computation times even further.

Another proposed approach was to adjust route-choice in a first stage, either off-line or in real-time, and account for it when adjusting signal timing. In [5] drivers are informed about prevailing conditions and undertake the shortest path updated on a node by node basis. A decentralized controller then computes signal plans based on route-choice information captured through turning ratios. In [19] routes between each OD are enumerated and the proportion of the demand in each route are assigned based on the path-size logit model [20]. A traffic plan is obtained through a multiobjective approach considering concomitantly throughput, delay, and traffic safety through a rolling-horizon approach. The multiobjective concept was also followed by [21] where they develop a reinforcement learning approach for real-time traffic signal control.

Compared to the macroscopic models used so far, [22] followed a rather different approach. The well known space-time network model for routing is used along with an analogous phase-time network model proposed by the own authors, leading to a so called space-phase-time hypernetwork model formulation. The resulting model is roughly a microscopic model with each vehicle being an agent and having a detailed representation of intersections. Reliable two-way communication is assumed between vehicles and infrastructure. The original formulation is decomposed in two sub-problems, one for traffic signal and the other for routing, by the use of Lagrangian relaxations so as to reduce the computational time required for optimization. Still, the decomposed formulation retained explicit coupling opposed to some previous strategies that relied on decoupling (or *ad-hoc* coupling). The solution of the problem employs a dynamic programming algorithm specifically tailored for the decomposed problem and is suitable for use in a parallel computing setting. The time required for the solution is in the order of minutes for a grid network with nine intersections. Nevertheless, it represents quite an achievement for such a detailed model.

In this work, we derive the controller based on a multi-commodity macroscopic traffic flow model similar to the multi-destination approach by [4]. This work is distinguished by the fact that our model can be cast into a convex optimization problem which allows us to obtain optimal signal times and route information without resorting to heuristics as done by [4]. This also allows the application of the method in larger networks. Compared to approaches based on the back-pressure algorithm [16], [17], the method proposed here considers the entire network when computing control actions and coordinates accordingly, thereby avoiding locally optimal policies that may not improve the system performance.

III. MULTI-COMMODITY SIGNAL CONTROL AND ROUTING

In this section we propose a real-time traffic management policy that integrates multi-commodity traffic signal control and routing of connected vehicles in multi-destination vehicular *ad-hoc* networks, through a novel multi-commodity formulation of the store-and-forward modeling paradigm.

A. Multi-Commodity Store-and-Forward Modelling

In the store-and-forward modeling paradigm (see, e.g., [8], [9]) each road segment is modeled as a queue of vehicles (the *store* part) and discharged vehicles are transferred directly to downstream queues (the *forward* part). The queues are bounded and vehicles are not able to join a link if its associated queue is full. This model captures the main aspects of urban traffic such as the initiation of congestion (queues grow if the incoming flow exceeds the maximum outflow), propagation (if the queue reaches its maximum, it blocks the flow in the upstream roads), and dissipation (queues decrease if the incoming flow is lower than the prevailing outflow).

The mathematical notation in this article is summarized in Table I. The urban network is represented by a directed graph defined by a set of links, Z, and a set of intersections, J. A link $z \in Z$ is a directed connection between two intersections. The formulation is discrete in time, with time

TABLE I
MATHEMATICAL NOTATION.

Symbol	Description
$a_z^d(k)$	demand originated at link z heading to destination d
$\alpha, \beta, \gamma, \rho$	weights on the penalty terms of the objective function
C	cycle time
D	set of destination links; indexed by $d \in D$, with $D \subseteq Z$
F_j	set of stages at junction j; indexed by $i \in F_j$
$g_{j,i}(k)$	green time of stage i at intersection j
$g_{j,i}^{\text{fix}}$	fixed green time of stage i at intersection j
$g_{j,i}^{\min}$	minimum green time for stage i at intersection j
$G_{z,m}^d(k)$	effective green time given to vehicles at link z heading to destination d taking downstream link m
I_j	set of incoming links of intersection j
J	set of intersections; indexed by $j \in J$
$\mathcal{J}$	objective function
$\mathcal{J}_{var}$	objective function terms; $var \in \{\mathbf{x}, \text{g}, \text{G}, \text{RDT}\}$
j_z^{D}	downstream intersection of link z
j_z^{U}	upstream intersection of link z
k	discrete step index; $k \in \{0, 1, \ldots, K\}$
K	horizon length in number of steps; $K \in \mathbb{N}_0$
L_j	lost time at junction j
$N(z, d)$	distance between links z and d
O_j	set of outgoing links of intersection j
$q_{\text{in},z}^d(k)$	inflow into link z of vehicles heading to destination d
$q_{\text{out},z}^d(k)$	outflow from link z of vehicles heading to destination d
$q_{z,m}^d(k)$	outflow from link z of vehicles heading to destination d taking downstream link m
s_z	saturation flow rate of link z
S_{Z}	set of signalized links; $S_{\text{Z}} \subseteq Z$
t	time
$t_{z,m}^{\text{fix},d}$	fixed fraction of vehicles at link z heading to destination d taking downstream link m
$t_{z,m}^{\text{op},d}$	fraction of vehicles at link z heading to destination d taking downstream link m
T	discrete-time step size
V_z	set of stages that give right of way (r.o.w.) to link $z \in I_j$; $V_z \subseteq F_j$
$x_z(k)$	total number of vehicles in link z
$x_z^d(k)$	number of vehicles on link z heading to destination d
$x_z^{\max}$	storage capacity of link z in number of vehicles
$x_z^{\text{V2I},d}(k)$	number of vehicles on link z heading to destination d obtained via V2I communication
Z	set of links; indexed by $z \in Z$ or $m \in Z$

Variables indexed by k refer to their value at time step k.

$t = kT$, $k \in \{0, 1, \ldots, K\}$ the discrete time step, T the time step length, and K the horizon length in number of steps. Thus, the horizon is divided in $K \in \mathbb{N}_0$ steps and the problem is defined from the present moment, $t = kT$ at $k = 0$ until the horizon length, $t = KT$ at $k = K$. While the store-and-forward model can be used with different time-steps [8], we follow [9] and assume that the time-step T equals the traffic cycle length C, i.e., $T = C$. Moreover, the cycle time is the same for the whole network (or regions of the network), so as to enable coordination.

The green times $g_{j,i}$ of stage i at intersection j are constrained by the cycle time C and lost time L_j[1]:

$$\sum_{i \in F_j} g_{j,i}(k) + L_j = C, \tag{1}$$

with F_j the set of stages in intersection $j \in J$. Moreover, stages are subject to a minimum green time $g_{j,i}^{\min}$:

$$g_{j,i}(k) \geq g_{j,i}^{\min}, \quad i \in F_j, \ j \in J. \tag{2}$$

[1]*Notation:* Here and in the rest of the paper all equations are assumed to hold for all $k = 0, \ldots, K - 1$, unless otherwise stated.
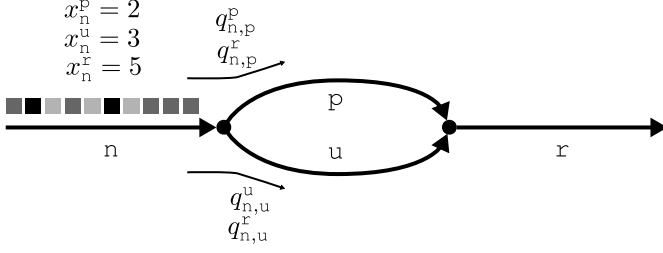
Fig. 1. Multi-commodity representation of a simple diverge-merge network.

The proposed multi-commodity approach considers different vehicle types traveling in a single link based on given properties or characteristics. To this end, each possible destination is considered as a different commodity, i.e., vehicles are classified by their intended destination $d \in D \subseteq Z$, with D the set of destination links, thus enabling routing.

The key relationship of the store-and-forward model is the conservation of vehicles. In the proposed multi-commodity formulation, the flow balance in a link is segmented by commodity. Precisely, the number of vehicles heading to destination d on link z, x_z^d, evolves as:

$$x_z^d(k+1) = x_z^d(k) + C\left[q_{in,z}^d(k) + a_z^d(k) - q_{out,z}^d(k)\right], \quad (3)$$

with $q_{in,z}^d(k)$, $q_{out,z}^d(k)$, $a_z^d(k)$, respectively, the inflow, the outflow, and the demand originated on link z heading to destination d at time step k. Note that a_z^d is assumed to be known from historical data or obtained/estimated in real-time from vehicles via V2I communication.

Inflows and outflows at every link z are also determined per commodity. For vehicles that have not reached their destinations ($z \neq d$), the outflows are driven by the effective green time, $G_{z,m}^d$ (link phase), on link z for vehicles heading to destination d that are taking downstream link m:

$$q_{z,m}^d(k) = \frac{G_{z,m}^d(k)s_z}{C}, z \in S_Z, z \neq d, m \in O_j, j = j_z^D, \quad (4)$$

where s_z is the saturation flow in link z; $S_Z \subseteq Z$ is the set of signalized links in the network; $O_j \subset Z$ is the set of outgoing links of intersection $j \in J$; and $j_z^D \subset J$ is the downstream intersection of link z.

Figure 1 illustrates the state variables and flows for a simple diverge-merge network. Vehicles, represented by coloured boxes, in link n are heading to three different destinations: p (dark vehicles), u (light gray vehicles), and r (dark gray vehicles). The corresponding state variables x_n^d are annotated in the figure. The flows are specific to each commodity and downstream link. Therefore, all vehicles heading to p (u) are properly routed to that link via flow $q_{n,p}^p$ ($q_{n,u}^u$). Vehicles heading to r have alternative routes, and are partly routed through p and partly through u via flows $q_{n,p}^r$ and $q_{n,u}^r$, respectively. Flow variables with zero value are not shown in the figure. Eventually these vehicles will merge into link r. Traffic signal timings (not shown) are computed accordingly.

For vehicles that have arrived at their destination ($d = z$), it is assumed that they leave the network at the same time

step. Therefore:

$$q_{out,z}^d(k) = \begin{cases} x_z^d(k), & \text{if } z = d, \\ \sum_{m \in O_j} q_{z,m}^d(k), & \text{if } z \neq d, z \in S_Z, d \in D, j = j_z^D, \\ 0, & \text{if } z \notin S_Z, z \neq d, \end{cases}$$
$$(5)$$

where $z \in Z$, and $d \in D$. The inflows are computed by:

$$q_{in,z}^d(k) = \sum_{i \in I_j} q_{i,z}^d(k)$$
$$= \frac{1}{C}\sum_{i \in I_j} s_i G_{i,z}^d(k), \quad z \in Z, j = j_z^U, d \in D, \quad (6)$$

where $I_j \subset Z$ is the set of incoming links of intersection $j \in J$, and $j_z^U \subset J$ is the upstream intersection of link z. The effective green times, $G_{z,m}^d$, cannot exceed the total stage times and are always non-negative, thus:

$$G_{z,m}^d(k) > 0, \quad z \in S_Z, \ d \in D, \ m \in O_j, \ j = j_z^D,$$
$$\sum_{m \in O_j, \ d \in D} G_{z,m}^d(k) \leq \sum_{i \in V_z} g_{j,i}(k), \quad z \in S_Z, \ j = j_z^D, \quad (7)$$

with $V_z \subseteq F_j$ the set of stages that give right of way (r.o.w.) to link $z \in I_j$.

Although vehicles are divided into different commodities, the storage capacity at link z, x_z^{max}, is shared among all commodities within the link, that totals to x_z. All movements receive r.o.w. in the same stage for a given link. In the case of a different stage set for a left-turn movement having right of way at different stages, the left pocket can be modeled as a different link. The state variables are bounded as follows:

$$x_z^d(k) > 0, \quad z \in Z, \ d \in D,$$
$$x_z(k) = \sum_{d \in D} x_z^d(k) \leq x_z^{max}, \ z \in Z. \quad (8)$$

The key feature of the pursued multi-commodity approach, compared to single-commodity formulations (see, e.g., [8], [9]), is that the destination turning rates at the network links are control variables, thus downstream inflows are not computed with respect to constant/pre-specified turning movement rates.

We assume that offsets and cycle times are constant or can be obtained in parallel by other algorithms as in [7].

B. Integrated Signal Control and Routing via Rolling Horizon

Next, an optimal control problem cast into a rolling horizon structure is presented with the prediction model, the objective function, and the control and prediction horizons.

1) Prediction Model: the multi-commodity formulation of the store-and-forward model proposed in the previous section serves as a prediction model and is used to solve the integrated signal control and routing by a rolling horizon approach [23]. The control and prediction horizons are assumed to be equal. In general, the horizon length is a trade-off depending on different factors including the network size, demand, and

computational effort. With an accurate model for prediction, a longer horizon improves performance at the expense of higher computation times. On the other hand, if the model is not as accurate, a longer horizon can lead to an erratic behavior due to the mismatch between the predicted and the realized states. In this case, the controller plans actions for the following time steps that will prove inadequate in the future. In turn, this leads to another sequence of actions to correct the previous inadequate actions that might also be inadequate and so on.

2) Objective Function: the objective function aims to reduce the risk of over-saturation and spill-back, so that $x_z(k) \leq x_z^{\max}$ following [9]. However, we also add to the cost function a terminal cost as well as a few penalty terms to overcome some model limitations. The objective function reads:

$$\sum_{k=0}^{K} \left(\sum_{\forall z \in Z} \frac{x_z(k)^2}{x_z^{\max}} + \rho \mathcal{J}_{\mathrm{x}}(k) \right) + \beta \mathcal{J}_{\mathrm{g}} + \gamma \, \mathcal{J}_{\mathrm{G}} + \alpha \mathcal{J}_{\mathrm{RDT}}, \quad (9)$$

with $\mathcal{J}_{\mathrm{RDT}}$ the terminal cost defined as:

$$\mathcal{J}_{\mathrm{RDT}} = \sum_{z \in Z} \sum_{d \in D} N(z, d) x_z^d(K), \quad (10)$$

where $N(z, d)$ is the distance between link z and d, the only term in the objective that weights differently each commodity. The penalty terms $\mathcal{J}_{\mathrm{x}}(k)$, $\mathcal{J}_{\mathrm{g}}$, $\mathcal{J}_{\mathrm{G}}$ are defined as follows:

$$\mathcal{J}_{\mathrm{x}}(k) = \max_z \left(\frac{x_z(k)}{x_z^{\max}} \right), \; z \in Z, \quad (11)$$

$$\mathcal{J}_{\mathrm{g}} = \sum_{j \in J, i \in F_j} \max_k \left(|g_{j,i}(k+1) - g_{j,i}(k)| \right), \quad (12)$$

$$\mathcal{J}_{\mathrm{G}} = \sum_{\substack{z \in S_Z, d \in D, \\ m \in O_j, j = J_z^{\mathrm{D}}}} \max_k \left(|G_{z,m}^d(k+1) - G_{z,m}^d(k)| \right). \quad (13)$$

The terminal cost $\mathcal{J}_{\mathrm{RDT}}$ is introduced in order to weight potential costs in the period $[K + 1, \infty)$ based on the state of the system at the final step of the prediction horizon, i.e., at $k = K$. Prescribing the terminal cost adequately allows the controller to achieve satisfactory performance with a shorter prediction horizon [23]–[25]. Observe from (10) that the product $N(z, d) x_z^d(K)$ is the expected distance that vehicles at link z heading to destination d have to travel to reach their destination. Since the goal is to bring the vehicles to their destinations, we define the terminal cost based on the remaining distance to be traveled (RDT). This encourages the controller to plan the actions so as to reach step K with vehicles as close as possible to their intended destinations.

The penalty terms $\mathcal{J}_{\mathrm{x}}$, $\mathcal{J}_{\mathrm{g}}$, $\mathcal{J}_{\mathrm{G}}$ are added to mitigate two key modelling challenges that would, otherwise, require complex non-linear formulations leading to non-convex optimization.

The first challenge concerns the holding back problem [4], [26], [27], which refers to the violation of the flow maximization principle. This principle states that drivers attempt to drive at maximum speed as long as it does not violate any constraint imposed by the prevailing traffic conditions, meaning that one or more upper or lower constraints will be binding. However, in the pursued multi-commodity formulation a control

action $G_{z,m}^d(k)$ is possible such that $\sum_{m \in O_j, \, d \in D} G_{z,m}^d(k) < \sum_{i \in V_z} g_{j,i}(k)$ (see (7)), $q_{\mathrm{out},z}^d(k) < x_z^d(k)$ (i.e., there is enough demand to discharge more vehicles), and $x_m < x_m^{\max} \; \forall m \in O_j$ (i.e., all downstream links can accept more vehicles). In other words, this control action attempts to hold vehicles in link z in a situation where vehicles have r.o.w. (i.e., green-time) and enough storage space in the downstream link to proceed. In case of congested traffic this situation may occur in the model, so as to avoid queue spill back further downstream. However, this would not happen in practice as vehicles would be able to proceed.

The second challenge concerns the first-in-first-out (FIFO) discipline across different commodities [11], [28]. In the pursued multi-commodity modelling, the green-time allocated to link z, $\sum_{i \in V_z} g_{j,i}(k)$, can be freely distributed to any combination of $G_{z,m}^d$ regardless of the order they entered in z, or of the current state, x_z^d. For instance, consider a single-lane link z where there are $x_z^{d_1}$ and $x_z^{d_2}$ vehicles heading to destinations d_1 and d_2, respectively, queued in random order. Based on the multi-commodity model, it could be beneficial to first discharge vehicles heading to d_1 and afterwards discharge the vehicles heading to d_2. Again, this would not happen in practice because, roughly, vehicles must be discharged in the same order they entered the link.

When assessing control actions of the rolling horizon controller for different initial conditions we observed two common occurrences during simulations that revealed these limitations. Occasionally, priority was given to one commodity on the first steps of the horizon, whereas the other commodity was discharged in later steps of the horizon, characterizing a first-in-first-out violation. The holding back was observed in occasions in which one link yielded low outflow during the first steps of the horizon, leading to higher queues, followed by a sudden increase of outflow in the same link on later steps of the horizon. Therefore, both effects are linked to the following: (i) high variation of flows in a given link for a given commodity along the prediction horizon due to both FIFO violation (i.e., high flow when a commodity is prioritized; low otherwise) and the holding back problem; (ii) large queues along with low flows due to the holding back problem.

The approach to mitigate these problems was to penalize these two occurrences in the objective function by discouraging high variation of flows along the horizon and also for large queues. Specifically, we define the penalty on variations of effective green times, denoted as $\mathcal{J}_{\mathrm{G}}$, and stage times, denoted $\mathcal{J}_{\mathrm{g}}$. We also added a penalty on the value of the maximum queue, $\mathcal{J}_{\mathrm{x}}$, since the holding back is also associated to large queues on the first steps of the prediction horizon. Note that $\mathcal{J}_{\mathrm{x}}$ represents a penalty on the longest queue at time step k, thereby avoiding the risk of queue spill backs specially when it is not avoided by the terms in the inner summation of (9). These three measures proved effective in mitigating the described challenges in macroscopic traffic flow modelling.

3) Overall Formulation: The control actions are obtained from the solution of an optimal control problem involving the objective function (9), the conservation equation (3), model constraints (1), (2), (4)–(8), and the given initial state. The

complete formulation of the control problem reads:

$$\min_{G_{z,m}^d, g_{j,i}} \quad \mathcal{J} = \sum_{k=0}^{K} \left(\sum_{\forall z \in Z} \frac{x_z(k)^2}{x_z^{\max}} + \rho \mathcal{J}_{\mathrm{x}}(k) \right) + \beta \mathcal{J}_{\mathrm{g}}$$
$$+ \gamma \, \mathcal{J}_{\mathrm{G}} + \alpha \mathcal{J}_{\mathrm{RDT}}$$

subject to: (1)–(8),
$$x_z^d(0) = x_z^{\mathrm{V2I},d}(0) \text{ is given } \forall z \in Z, d \in D,$$
$$a_z^d(k) \text{ is given } \forall z \in Z, d \in D. \qquad (14)$$

where $x_z^{\mathrm{V2I},d}(0)$ is the current state of the network obtained via V2I communication (see Section III-C for more information).

The mathematical program (14) corresponds to a convex quadratic programming problem with linear constraints. Convexity ensures that the problem is computationally tractable even for large-scale networks involving thousand of variables and constraints. Nevertheless, to speed up the computational time, we transform the quadratic cost $x_z^2(k)/x_z^{\max}$ into a piecewise linear function by means of separable programming [29], so that the problem can be solved by linear programming with simplex or interior point methods [30]. Also, the max operator in $\mathcal{J}_{\mathrm{x}}(k)$, $\mathcal{J}_{\mathrm{g}}$, and $\mathcal{J}_{\mathrm{G}}$ are implemented by creating an auxiliary variable and adding a constraint for each element within the max operator and then adding the auxiliary variable into the objective. We refer to [29] for further details.

The control is implemented via a rolling horizon scheme [23]. At a given control time step k_0, the initial state, $x_z^d(k_0)$, and demand predictions $a_z^d(k_0), \ldots, a_z^d(k_0 + K - 1)$ are obtained. The controller computes the optimal control sequence for the next $K - 1$ steps, i.e., from step k_0 to $k_0 + K - 1$, as the solution of (14). Only the first step of the control sequence is implemented. In the next time step, $k_0 + 1$, the same procedure is performed shifted one step in time, i.e., from step $k_0 + 1$ to $k_0 + 1 + K - 1$, and only the first element of the new control sequence is implemented, and so forth.

C. Implementation and Tuning

For the solution of the convex optimization problem (14) via rolling horizon, reliable information on the arrival rates a_z^d and initial states $x_z^d(0)$ at every $T = C$ is needed. Assuming V2I communication, arriving vehicles at link z will be able to report their final destination in order for the infrastructure to gather information on the arriving flows, a_z^d, and number of vehicles, $x_z^{\mathrm{V2I},d}(0)$. Reporting of the final destination can be done only once or whenever the final destination $d \in D$ is changed. In any case the infrastructure will be able to know in real-time the number of vehicles within each link of the network via V2I. If this is not the case, inductive-loop detectors and historical information (e.g., OD data) can be used to observe the state of the system via appropriate extrapolation and estimation methods (see, e.g., [9]).

The routing information should be transmitted from the infrastructure to the individual vehicles via V2I. We assume that all drivers follow the suggested route. To this end, we propose a scheme to operationalize routing information to individual vehicles with the downstream link being provided and updated upon entrance in a new link. Precisely, we define

operational turning rate probabilities at every link z based on the effective greens resulting from the solution of (14):

$$t_{z,m}^{\mathrm{op},d} = \frac{q_{z,m}^d(0)}{\displaystyle\sum_{i \in O_j} q_{z,i}^d(0)}, \quad j = j_z^{\mathrm{D}}. \qquad (15)$$

Every vehicle entering link z heading to destination d will be routed through link m with probability $t_{z,m}^{\mathrm{op},d}$. This probability remains the same until the next control action is computed on the next traffic cycle. Since our strategy does not require high frequency message exchanges—the vehicle communicates its destination and the road side unit responds informing the next link of the vehicle path, we do not impose any specific type of communication infrastructure. Essentially, the communication between a vehicle and the infrastructure is required at every link transition. Therefore, a practical implementation would require a road side unit at every intersection or at every link with the communication range spanning the link length.

The proposed multi-commodity formulation brings along three additional weights β, γ, and ρ on the penalty terms, and the weight α on the terminal cost. The (control and prediction) horizon K is also a key tuning parameter.

The basic guideline for adjusting the parameters α, β, γ, and ρ is to first set them to zero, solve the problem, and then adjust their values based on a qualitative analysis of the results by recording all initial states and the controller outputs along the horizon. Then, repeat this process with the new values until a satisfactory result is reached. Some guidelines are:

1) α should be increased whenever the flows for all or most commodities in the last step of the horizon are substantially smaller than in the earlier steps;
2) β and γ should be increased if the computed control actions; show variations over the prediction horizon;
3) ρ should be increased whenever large queues are observed in a few links whereas substantially smaller queues occur in neighboring links.

The penalties on high variations along the horizon are designed to mitigate the FIFO violation and holding back problems. The penalty on the largest queues attempts to avoid the queue spillback in critical links.

Finally, the adjustment of the prediction horizon also relies on a trial-and-error procedure. In the absence of modeling errors, the horizon needs to be chosen as a trade-off of CPU-time (which increase as K increases) and performance (generally higher as K increases). Since the proposed formulation is a coarse approximation of the actual dynamics, a better performance is not necessarily expected as the horizon K grows. On the other hand, the horizon should be long enough for the controller to be sensitive to the impact of vehicles reaching their destination, thereby reducing x_z^d in all links and, as a consequence, also reducing the objective function. For the proposed formulation, this would be guaranteed if the horizon, K, is equal to the number of links of the longest OD pair.

D. Alternative Formulations

The integrated multi-commodity approach can be adapted to work as only signal control or only routing.

1) Signal Control Only: the single-commodity traffic signal strategy [9] assumes turning rates are fixed or known beforehand. In the multi-commodity case, we can benefit from information obtained via V2I communication and assume destination specific turning rates, which can be fixed or estimated in real-time. Assuming fixed turning rates, $t_{z,m}^{\text{fix},d}$, we can add:

$$q_{z,m}^d(k) = t_{z,m}^{\text{fix},d} q_{\text{out},z}^d(k), \ z \in S_\text{Z}, \ d \in D, \quad (16)$$

to the problem constraints.

In terms of control variables, the signal control-only case is similar to the one by [9] in which the ultimate control variables are the stage green-times and phases with the difference being the use the information about the vehicles destination. Potentially, the use of more detailed information can lead to better performance. However, the multi-commodity formulation leads to significantly higher computational times compared to the single-commodity formulation. Nevertheless, the problem is still convex and mathematically tractable.

2) Routing Only: a route control can also be derived from this model. In this case the controller can still change the commodity green times, $G_{z,m}^d$, but cannot change the stages green time, $g_{j,i}$. Similar to the previous case, assuming fixed stage times, $g_{j,i}^{\text{fix}}$, equality constraints are added on the stages:

$$g_{j,i}(k) = g_{j,i}^{\text{fix}}, \ j \in J, \ i \in F_j. \quad (17)$$

In that case, the controller is able to change the commodity green, $G_{z,m}^d$ as long as (7) holds.

IV. Application and Results

We evaluate the proposed control strategy with microscopic simulator Aimsun [31]. Three sizes of a grid network are simulated with time varying-demands. The results are compared to the single-commodity approach proposed by [9].

A. Grid Networks

The three modelled networks are depicted in Fig. 2. Nodes represent intersections, and arcs represent links and their respective directions. Each link label is obtained by the concatenation of the labels of the nodes it connects, including labels from sink and source nodes (not in the figure). Dashed perimeters delimit each network (Fig. 2), referred to as large (L) with 20 intersections, 9 origins and 9 destinations (black area); medium (M) with 16 intersections, 8 origins and 8 destinations (dark gray area); and small (S) with 12 intersections, 7 origins and 7 destinations (light gray area).

All intersections are signalized, each with two incoming links and two outgoing links. All links are single lane with length $l = 500$ m, and a saturation flow rate $s = 2000$ veh/h. Each incoming link is served by one stage; the first stage gives r.o.w. to horizontal links in the figure and the second stage to the vertical ones. The minimum stage time is 20 s, the yellow time is 3 s, and the clearance red is 2 s. The default simulator values were kept for all driving behavior parameters.

B. Demand

We designed a time-varying traffic demand that allows for the demonstration of the effect of the controller on signal
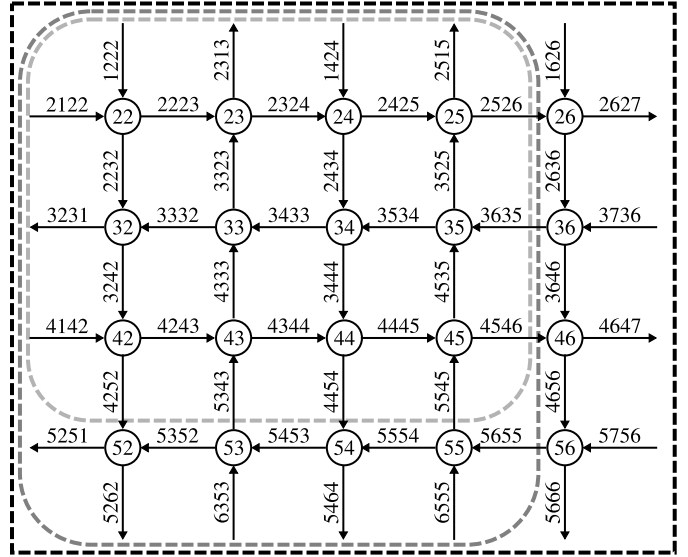

Fig. 2. Modeled grid networks: large (L) - 20 intersections, 9 origins and 9 destinations (black area); medium (M) - 16 intersections, 8 origins and 8 destinations (dark gray area); and small (S) - 12 intersections, 7 origins and 7 destinations (light gray area).

times and routing of vehicles (Table II). The Dynamic Traffic Assignment feature of Aimsun was disabled for all simulations. When routing was enabled, vehicles were routed on a link-by-link basis as computed by the control strategy, otherwise destination-specific turning rates were defined offline for each link based on pre-defined routes. For some routes, the demand is slightly above capacity. However, a routing for which the demand can be supplied exists in all cases.

Besides the origin-destination (OD) pairs with only through movements (e.g., OD pairs (2122, 2627), (1222, 5262), etc.) that have a constant flow of 400 veh/h, six more OD pairs with time-varying demands were defined in time slices of 20 min (Table II).

Figure 3(a) depicts the possible routes for OD pairs (2122, 4647), (3736, 2313), (6353, 3231) by bold coloured links. The first OD pair has three possible routes (all with the same length), while the other two have only one possible route each. The highlighted nodes show the intersections that have their capacity affected by the combination of the flows in the routes of OD pair (2122, 4647) with OD pairs (3736, 2313) and (6353, 3231), thus calling for new signal timings and rerouting of vehicles in OD pair (2122, 4647) when real-time control is enabled. Figure 3(b) sketches the demand profiles for the three OD pairs (not routes) for time 0 to 40 min. Clearly, the high demands for OD pairs (3736, 2313) and (6353, 3231) occur for different periods of time, thus giving the opportunity of routing vehicles in OD pair (2122, 4647) through different intersections (with different signal timings) in the two periods of time when under real-time control. Analogous effect is produced by the flows in remaining OD pairs in Table II.

For the scenarios without real-time routing the flows were split evenly between alternative routes with equal lengths for a given OD pair (e.g., the three routes of OD pair (2122, 4647) in Fig. 3(a)). We acknowledge that this route assignment is simple, but deemed fair under real-time signal control. However, it is not the purpose of this article to find the

TABLE II

DEMAND (VEH/H) FOR ALL ORIGIN-DESTINATION (OD) PAIRS IN THE THREE SIMULATED GRID NETWORKS.

| OD Pair | | | 0–20 | 20–40 | 40–60 | 60–80 | 80–100 | 100–120 |
L (black)	M (dark gray)	S (light gray)	min	min	min	min	min	min
(2122, 4647)	(2122, 4546)	(2122, 4546)	600	600	50	50	0	0
(6353, 3231)	(6353, 3231)	(5343, 3231)	600	50	50	50	0	0
(3736, 2313)	(3635, 2313)	(3635, 2313)	50	600	50	50	0	0
(6353, 2627)	(6353, 2526)	(5343, 2526)	50	50	600	600	0	0
(3736, 5464)	(3635, 5464)	(3635, 4454)	25	25	600	25	0	0
(1424, 3231)	(1424, 3231)	(1424, 3231)	25	25	25	600	0	0
OD pairs connected only by through movements			400	400	400	400	400	400
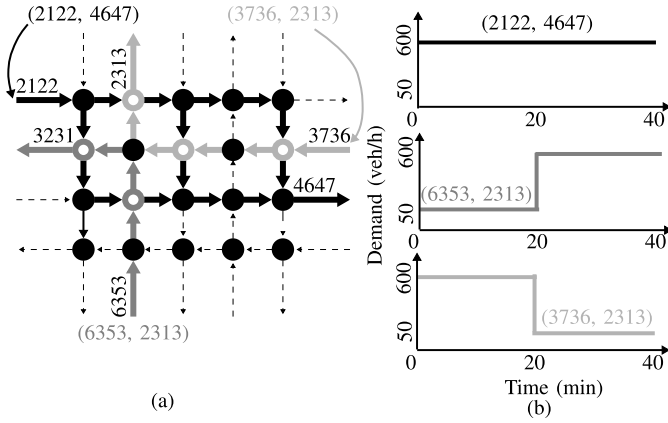
Fig. 3. (a) Possible routes for vehicles of OD pairs (2122, 4647) in bold black, (6353, 3231) in dark gray, and (3736, 2313) in light gray, with affected intersections highlighted in gray colors; and (b) sketch of the demands for the same OD pairs.

optimal timing considering route choice dynamics under User Equilibrium (UE). Rather, the purpose is to assess whether the controller will respond adequately for changes in real time. As a matter of fact, traffic signal control strategies at large, implicitly or explicitly, assume fixed routes. Since the signal times change dynamically, the UE is not known beforehand. Achieving Dynamic User Equilibrium is challenging due to transient nature of traffic, especially when considering a relatively small control horizon [5].

C. Control Strategies

Three control strategies were tested for each network size: the rolling-horizon controller proposed by [9] based on the single-commodity formulation (QPC), and the rolling-horizon controllers based on the multi-commodity formulation with only signal enabled (MCS) and with integrated signal control and routing (MCR). The cycle time equals 100 s and is fixed for all intersections in all control scenarios. The offsets were optimized offline using Transyt-10 [32].

The GLPK LP solver [33] was used for obtaining the control action for the multi-commodity approaches (MCS and MCR) and CVXOPT [34] was used to compute the control actions for the single-commodity case (QPC).

Results are reported for horizons of 2, 5, and 8 cycles. In the QPC case there are no adjustable parameters. For the MCS and MCR cases the parameters were manually adjusted following the basic guidelines outlined in Section III-C for each case, since the weights are sensitive to the number of commodities

TABLE III

RESULTS FOR SINGLE-COMMODITY STORE-AND-FORWARD (QPC), AND MULTI-COMMODITY WITH SIGNAL CONTROL ONLY (MCS) AND WITH INTEGRATED SIGNAL CONTROL AND ROUTING (MCR) FOR HORIZONS K OF 2, 5, AND 8 CYCLES. THE PERCENTAGES ARE RELATIVE TO THE QPC CASE WITH $K = 2$ CYCLES AND PRIOR TO DATA ROUNDING-OFFS.

| Size | K | Ctrl. | Delay | % | TTS | % | Avg. sp. | % |
			(cycs.)	(s/km)		(veh·h)		(km/h)
		QPC	55	0	541	0	30	0
	2	MCS	70	27	607	12	26	−11
		MCR	53	−4	535	−1	30	2
		QPC	58	5	552	2	29	−2
S	5	MCS	57	2	547	1	29	−1
		MCR	47	−16	504	−7	32	8
		QPC	64	15	574	6	28	−6
	8	MCS	60	8	565	4	29	−3
		MCR	50	−9	523	−3	31	4
		QPC	56	0	655	0	30	0
	2	MCS	65	1	707	8	27	−7
		MCR	54	−3	644	−2	30	1
		QPC	58	5	668	2	29	−2
M	5	MCS	57	2	660	1	29	−1
		MCR	49	−11	621	−5	31	5
		QPC	62	11	684	5	28	−5
	8	MCS	58	3	665	2	29	−1
		MCR	51	−9	626	−4	31	4
		QPC	47	0	794	0	32	0
	2	MCS	52	10	830	5	31	−4
		MCR	45	−4	783	−1	32	2
		QPC	48	2	800	1	32	−1
L	5	MCS	46	−2	787	−1	32	1
		MCR	37	−21	722	−9	35	10
		QPC	51	8	821	3	31	−3
	8	MCS	47	0	796	0	32	0
		MCR	39	−17	736	−7	34	8

and horizon, resulting in α and γ in the range 1–10, β in the range 250–300, and ρ at around 50. Demand predictions were taken as the current value for the whole horizon, i.e., constant.

D. Results

The three grid networks (Section IV-A) were simulated with the designed demand (Section IV-B) for each of the selected control strategies (Section IV-C). The considered traffic performance metrics are: delay (s/km), space-mean speed (km/h), and total time spent (TTS) (veh·h) in the network. Table III summarizes the obtained results averaged over ten replications (simulation runs) with different seeds.

The results for the three networks are consistent despite their differences in size. The MCR strategy always outperforms the others strategies irrespective of the chosen horizon since it was able to manage the routes based on the incoming demand.
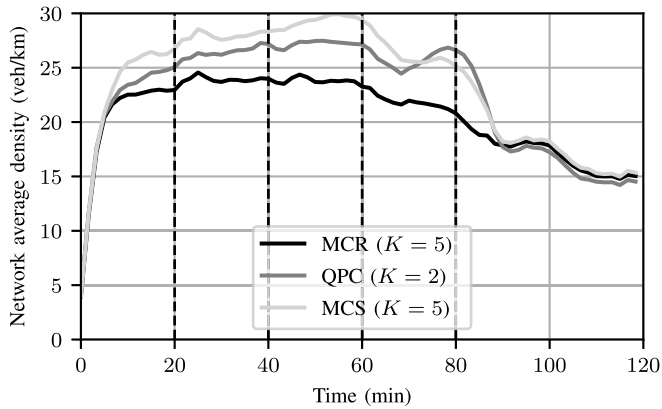
Fig. 4. Average, minimum and maximum densities over the 10 replications of the L network for MCR ($K = 5$ cycles), MCS ($K = 5$ cycles) and QPC ($K = 2$ cycles).
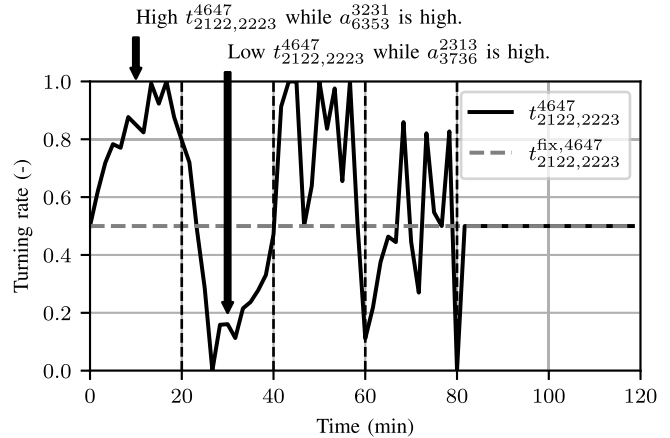

Fig. 5. Turning rates due to MCR routing and fixed turning rates from link 2122 to link 2223 for OD pair (2122, 4647) for one replication of the L network with $K = 5$ cycles.
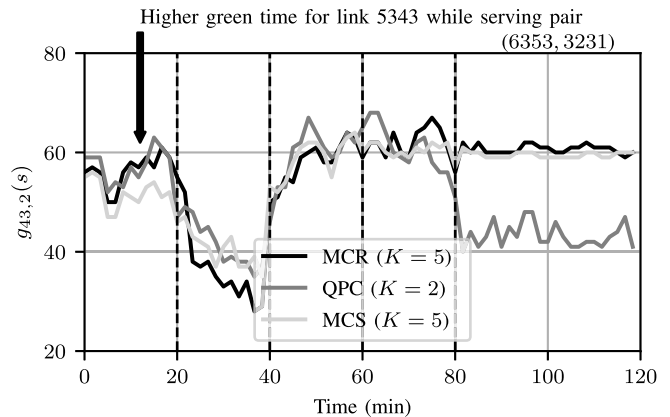

Fig. 6. Green time for stage 2 of link 5343 approaching intersection 43 for one replication of the L network due to MCR and MCS with $K = 5$ cycles, and QPC with $K = 2$ cycles.

When we consider each strategy alone and compare the results for the three simulated horizons, the best results of QPC occurred for $K = 2$ cycles (used as the base scenario) and the best results for MCS and MCR occurred for $K = 5$ cycles. For the horizons that lead to better results, the two strategies without routing (QPC and MCS) yield similar performance.

1) Average Density: Figure 4 depicts the network average densities of the best performance MCR ($K = 5$ cycles), MCS ($K = 5$ cycles) and QPC ($K = 2$ cycles) for the L (black) network. The vertical dashed lines indicate the time instants of demand changes (Table II). Observe that a slight increase in the network density occurs for the MCR strategy following the demand shifts. Nevertheless, the controller adapts and the network density reduces. Overall, the MCR strategy is able to keep lower density for most of the simulation. The lowest density is reflected by an increased throughput of MCR over QPC. Also note that once the demand drops, every strategy yields similar performance. This is expected since under low demands, the optimal routes are simply the shortest paths.

2) Routing: recall from Fig. 3 (Section IV-B) that it is expected that vehicles in the OD pair (2122, 4647) use routes that do not interfere with vehicles in the OD pair (3736, 2313) or in the OD pair (6353, 3231) when the demand in these pairs are high. Analogous reasoning apply for the other OD pairs.

The expected behavior was observed in all MCR cases. Figure 5 shows both the turning rates of vehicles from link 2122 to link 2223 towards destination 4647 for one replication of the L network with $K = 5$ cycles, and also the fixed turning rates for that specific turn. The vertical dashed lines highlight the instants of demand changes. Observe that in the first 20 minutes the demand to link 4647 is routed mostly through link 2223 so that it does not interfere with the high demand of OD pair (6353, 3231) that follows only one route with no alternatives. Once the demand of the OD pair (6353, 3231) ceases at time 20 min and the demand of the OD pair (3736, 2313) increases, the vehicles are then routed mainly through link 2232 (not shown). The turning rates varies significantly in the next 40 minutes, but it has little to no influence on the performance since the demand for pair (2122, 4647) is low at period. Regardless, one could

suppress the variations by penalizing the variations from the prevailing turning rates in the optimization problem.

3) Signal Timing: the changes in the routing would not be effective if the signal timings did not change accordingly. Figure 6 shows the stage time for the first stage of intersection 43 for the same cases discussed before for routing. In the first 20 minutes, the demand is high from OD pair (6353, 3231) therefore the flow is high through link 5343 (the only possible route for this OD pair). On the other hand, the flow on link 4243 is low, roughly equal to the through flow, since vehicles from OD pair (2122, 4647) are being routed through link 2223 (see Fig. 5). Therefore, the green time for link 5343 is high on the first 20 minutes. In the next 20 minutes, the demand from OD pair (6353, 3231) ceases and the demand from OD pair (2122, 4647) is routed through link 5343 and, thus the green time giving r.o.w to this link reduces significantly. Finally, the green time at that approach is high for the remainder of the simulation to serve the demand of OD pair (6353, 2627).

4) Computational Efficiency: We also measured the average CPU-time (s) necessary for the computation of the control sequences at each cycle for MCR, MCS and QPC. A PC with processor Intel(R) Core(TM) i5-45780 CPU 3.2 GHz clock under Ubuntu 17.10 operating system was used. The average computation times per cycle are depicted in Fig. 7.
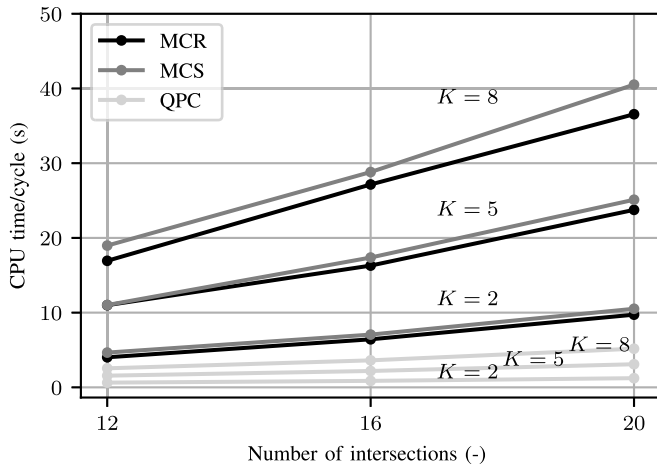
Fig. 7. Average computation times per cycle for the solution of the MCR, MCS and QPC optimization problems.

The two multi-commodity strategies have higher computational time because of the higher number of variables involved. The QPC strategy has one effective green-time per link per time-step and two inequality constraints for its upper and lower bounds. On the other hand, the MCR and MCS strategy have effective green-time for every combination of destination and downstream link which leads to a significantly higher number of variables and inequality constraints.

Nevertheless the CPU times are suitable for the intended application. Even for the larger prediction horizon considered ($K = 8$ cycles) and the larger network (20 intersections), it takes less than half of the cycle time ($C = 100$ s) to compute the control action both for MCR and MCS. Note that MCS requires a slightly higher computation time than MCR due the additional constraints. In practice, the control computation would start some forty seconds before the end of the current cycle so as to have the control actions available before the beginning of the next cycle.

V. Conclusion

A method for integrated signal control and routing of connected vehicles based on the multi-commodity store-and-forward model was presented. The rolling horizon controller based on the new formulation enabled handling routing in addition to signal timings. The method is specifically suitable for connected vehicles since it requires communication between the vehicle and the infrastructure, but no vehicle automation feature is required. The method is computationally efficient thanks to the convexity of the macroscopic modeling proposed.

Micro-simulation runs for grid networks of three sizes, varying demands and for different values of prediction horizon were performed to evaluate the method. The results showed that the proposed strategy outperforms the single-commodity QPC control by [9] and that it could effectively manage signal times and routing. The results indicate that the real-time application of the methods is practicable using standard equipment and open-source general solvers [33].

For future work, we will evaluate the performance of the proposed method in realistic networks with field observed

demands and routes obtained from available data sets such as [35]. Scenarios in which not all vehicles are connected should be investigated as well. In addition, more general methods to handle some of modelling challenges, especially with respect to first-in-first-out violation and holding back problem, could be investigated. Distributed networked control methods may be attempted to handle larger networks [36] as well as automated methods to ease the tuning of the control parameters [37].

References

[1] M. Papageorgiou, C. Diakaki, I. Nikolos, I. Ntousakis, I. Papamichail, and C. Roncoli, "Freeway traffic management in presence of vehicle automation and communication systems (VACS)," in *Road Vehicle Automation 2*, G. Meyer and S. Beiker, Eds. Cham, Switzerland: Springer, 2015, pp. 205–214.

[2] M. Rinaldi and C. M. J. Tampère, "An extended coordinate descent method for distributed anticipatory network traffic control," *Transp. Res. B, Methodol.*, vol. 80, pp. 107–131, Oct. 2015.

[3] M. Rinaldi, W. Himpe, and C. M. J. Tampère, "A sensitivity-based approach for adaptive decomposition of anticipatory network traffic control," *Transp. Res. C, Emerg. Technol.*, vol. 66, pp. 150–175, May 2016.

[4] Y. Han, A. Hegyi, Y. Yuan, C. Roncoli, and S. Hoogendoorn, "An extended linear quadratic model predictive control approach for multi-destination urban traffic networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 10, pp. 3647–3660, Oct. 2019.

[5] A. H. F. Chow, R. Sha, and Y. Li, "Adaptive control strategies for urban network traffic via a decentralized approach with user-optimal routing," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 4, pp. 1697–1704, Apr. 2020.

[6] D. C. Gazis, "Optimum control of a system of oversaturated intersections," *Oper. Res.*, vol. 12, no. 6, pp. 815–831, Dec. 1964.

[7] C. Diakaki *et al.*, "Extensions and new applications of the traffic-responsive urban control strategy: Coordinated signal control for urban networks," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 1856, no. 1, pp. 202–211, Jan. 2003.

[8] K. Aboudolas, M. Papageorgiou, and E. Kosmatopoulos, "Store-and-forward based methods for the signal control problem in large-scale congested urban road networks," *Transp. Res. C, Emerg. Technol.*, vol. 17, no. 2, pp. 163–174, Apr. 2009.

[9] K. Aboudolas, M. Papageorgiou, A. Kouvelas, and E. Kosmatopoulos, "A rolling-horizon quadratic-programming approach to the signal control problem in large-scale congested urban road networks," *Transp. Res. C, Emerg. Technol.*, vol. 18, no. 5, pp. 680–694, Oct. 2010.

[10] W. Kraus *et al.*, "Cost effective real-time traffic signal control using the TUC strategy," *IEEE Intell. Transp. Syst. Mag.*, vol. 2, no. 4, pp. 6–17, winter 2010.

[11] G. C. D'Ans and D. C. Gazis, "Optimal control of oversaturated Store-and-Forward transportation networks," *Transp. Sci.*, vol. 10, no. 1, pp. 1–19, Feb. 1976.

[12] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[13] J. Lei and U. Ozguner, "Combined decentralized multi-destination dynamic routing and real-time traffic light control for congested traffic networks," in *Proc. 38th IEEE Conf. Decis. Control*, Dec. 1999, pp. 3277–3282.

[14] J. Lei and U. Ozguner, "Integration of dynamic routing and intersection control in intelligent transportation system," in *Proc. ITSC. IEEE Intell. Transp. Syst.*, Oct. 2000, pp. 137–142.

[15] A. A. Zaidi, B. Kulcsar, and H. Wymeersch, "Back-pressure traffic signal control with fixed and adaptive routing for urban vehicular networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 8, pp. 2134–2143, Aug. 2016.

[16] Y. Liu, J. Gao, and M. Ito, "Back-pressure based adaptive traffic signal control and vehicle routing with real-time control information update," in *Proc. IEEE Int. Conf. Veh. Electron. Saf. (ICVES)*, Sep. 2018, pp. 1–6.

[17] H. Chai, H. M. Zhang, D. Ghosal, and C.-N. Chuah, "Dynamic traffic routing in a network with adaptive signal control," *Transp. Res. C, Emerg. Technol.*, vol. 85, pp. 64–85, Dec. 2017.

[18] T. Le, H. L. Vu, Y. Nazarathy, Q. B. Vo, and S. Hoogendoorn, "Linear-quadratic model predictive control for urban traffic networks," *Transp. Res. C, Emerg. Technol.*, vol. 36, pp. 498–512, Nov. 2013.

[19] X. Li and J.-Q. Sun, "Multi-objective optimal predictive control of signals in urban traffic network," *J. Intell. Transp. Syst.*, vol. 23, no. 4, pp. 370–388, Jul. 2019.

[20] M. C. J. Bliemer and P. H. L. Bovy, "Impact of route choice set on route choice probabilities," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 2076, no. 1, pp. 10–19, Jan. 2008.

[21] J. Jin and X. Ma, "A multi-objective agent-based control approach with application in intelligent traffic signal system," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 10, pp. 3900–3912, Oct. 2019.

[22] P. Li, P. Mirchandani, and X. Zhou, "Solving simultaneous route guidance and traffic signal optimization problem using space-phase-time hypernetwork," *Transp. Res. B, Methodol.*, vol. 81, pp. 103–130, Nov. 2015.

[23] J. A. Rossiter, *A 1st Course Predictive Control*, 2nd ed. Boca Raton, FL, USA: CRC Press, 2018.

[24] M. Ellis, H. Durand, and P. D. Christofides, "A tutorial review of economic model predictive control methods," *J. Process Control*, vol. 24, no. 8, pp. 1156–1178, Aug. 2014.

[25] M. Wang, W. Daamen, S. P. Hoogendoorn, and B. van Arem, "Rolling horizon control framework for driver assistance systems. Part I: Mathematical formulation and non-cooperative systems," *Transp. Res. C, Emerg. Technol.*, vol. 40, pp. 271–289, Mar. 2014.

[26] K. Doan and S. V. Ukkusuri, "On the holding-back problem in the cell transmission based dynamic traffic assignment models," *Transp. Res. B, Methodol.*, vol. 46, no. 9, pp. 1218–1238, Nov. 2012.

[27] A. K. Ziliaskopoulos, "A linear programming model for the single destination system optimum dynamic traffic assignment problem," *Transp. Sci.*, vol. 34, no. 1, pp. 37–49, Feb. 2000.

[28] M. Carey, H. Bar-Gera, D. Watling, and C. Balijepalli, "Implementing first-in–first-out in the cell transmission model for networks," *Transp. Res. B, Methodol.*, vol. 65, pp. 105–118, Jul. 2014.

[29] H. P. Williams, *Model Building in Mathematical Programming*. Hoboken, NJ, USA: Wiley, 2013.

[30] G. B. Dantzig, *Linear Programming and Extensions*. Princeton, NJ, USA: Princeton Univ. Press, 1998.

[31] J. Casas, J. L. Ferrer, D. Garcia, J. Perarnau, and A. Torday, "Traffic simulation with aimsun," in *Fundamentals of Traffic Simulation*, J. Barceló, Ed. New York, NY, USA: Springer, 2010, pp. 173–232, doi: 10.1007/978-1-4419-6142-6_5.

[32] M. Crabtree, R. Vincent, and S. Harrison, "Transyt/10 user guide," *TRL Appl. Guide AG*, to be published. [Online]. Available: https://trid.trb.org/Results?q=&serial=%22TRL%20APPLICATION%20GUIDE%20AG%2028%22#/View/481925

[33] A. Makhorin, *GNU Linear Programming Kit*, vol. 38. Moscow, Russia: Moscow Aviation Institute, 2001.

[34] M. Andersen, J. Dahl, and L. Vandenberghe. (2013). *CVXOPT: A Python Package for Convex Optimization*. [Online]. Available: https://cvxopt.org

[35] E. Barmpounakis and N. Geroliminis, "On the new era of urban traffic monitoring with massive drone data: The pNEUMA large-scale field experiment," *Transp. Res. C, Emerg. Technol.*, vol. 111, pp. 50–71, Feb. 2020.

[36] F. A. de Souza, E. Camponogara, W. Kraus, and V. B. Peccin, "Distributed MPC for urban traffic networks: A simulation-based performance analysis," *Optim. Control Appl. Methods*, vol. 36, no. 3, pp. 353–368, May 2015.

[37] A. Kouvelas, K. Aboudolas, E. B. Kosmatopoulos, and M. Papageorgiou, "Adaptive performance optimization for large-scale traffic control systems," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 4, pp. 1434–1445, Dec. 2011.

Rodrigo Castelan Carlson (Member, IEEE) received the B.Eng. degree in control and automation engineering and the M.E. degree in electrical engineering from the Federal University of Santa Catarina, Brazil, in 2004 and 2006, respectively, the B.B.A. degree in business administration from Santa Catarina State University, Brazil, in 2006, and the Ph.D. degree from the Technical University of Crete, Greece, in 2011.

From 2012 to 2015, he was an Assistant Professor with the Mobility Engineering Center, Federal University of Santa Catarina, where he was with the Department of Automation and Systems from 2015 to 2020 and has been an Associate Professor since 2020. His research interest includes control and optimization applications to traffic systems. In these areas, he has been serving as a reviewer for journals and conferences, as a consultant, as a member of program committees for conferences, and as a guest editor for journals.

Dr. Carlson was a recipient of the scholarship for postgraduate studies abroad from The Capes Foundation, Ministry of Education, Brazil, from 2007 to 2011, and the Gold Medal of the Young European Arena of Research 2010 in the pillar Mobility and Intermodality. In 2017, 2018, 2019, and 2020, he was also a recipient of the prize ANPET of Scientific Production. Since 2015, he has been a Researcher of The National Council for Scientific and Technological Development (CNPq). Since 2018, he has also been serving as an Associate Editor of *IET Intelligent Transport Systems Journal* and has also been a Review Editor of *Frontiers in Future Transportation* since 2020.

Eduardo Rauh Müller received the B.Eng. degree in control and automation engineering and the M.E. and Ph.D. degrees in systems engineering from the Federal University of Santa Catarina, Brazil, in 2011, 2013, and 2018, respectively.

From 2013 to 2015, he was a Consultant for state and municipal agencies in urban and freeway traffic and public transportation systems. Since 2018, he has been a Post-Doctoral Researcher with the Department of Automation and Systems, Federal University of Santa Catarina. His current research interests include control and optimization applied to traffic and transportation systems, and to oil production.

Konstantinos Ampountolas (Member, IEEE) received the Dipl.Ing. degree in production engineering and management, the M.Sc. degree in operations research, and the Ph.D. degree in engineering from the Technical University of Crete, Greece, in 1999, 2002, and 2009, respectively.

He was a Senior Lecturer with the James Watt School of Engineering, University of Glasgow, U.K., from 2013 to 2019, a Research Fellow with the École Polytechnique Fédérale de Lausanne, Switzerland, from 2012 to 2013, a Visiting Researcher Scholar with the University of California at Berkeley, Berkeley, CA, USA, in 2011, and a Post-Doctoral Researcher with the Centre for Research & Technology Hellas, Greece, in 2010. He was also a short-term Visiting Professor with the Technion–Israel Institute of Technology, Israel, in 2014, and the Federal University of Santa Catarina, Florianópolis, Brazil, in 2016 and 2019. Since 2019, he has been an Associate Professor with the Department of Mechanical Engineering, University of Thessaly, Greece. His research interest includes control and optimization with applications to transportation networks.

Dr. Ampountolas serves as an Associate Editor for the *Journal of Big Data Analytics in Transportation*, and on the editorial advisory boards for *Transportation Research Part C* and *Transportation Research Procedia*. He has also served as the Editor for the transportation of *Data in Brief* from 2018 to 2019.

Felipe de Souza received the B.Eng. degree in control and automation engineering and the M.E. degree in automation and systems engineering from the Federal University of Santa Catarina, Brazil, in 2008 and 2012, respectively, and Ph.D. degree in transportation systems engineering from the University of California at Irvine, Irvine, in 2018.

Since 2018, he has been a Postdoctoral Appointee with the Vehicle and Mobility Systems Group, Argonne National Laboratory. His current research interests include traffic modeling and control, transportation systems simulation, and dispatching algorithms for on-demand mobility services.

Dr. de Souza was recipient of the Science Without Borders Scholarship for postgraduate studies abroad from The CAPES Foundation, Ministry of Education, Brazil, from 2014 to 2018, and of the Best Paper Award at the IEEE CASE 2010 Conference. In addition, he was also a recipient of the 2017 ITS California Scholarship for continued research in Intelligent Transportation Systems.