IEEE Control Systems Letters paper presented at
2023 American Control Conference (ACC)
San Diego, CA, USA. May 31 - June 2, 2023

Optimization-based Coordination and Control of Traffic Lights and Mixed Traffic in Multi-Intersection Environments

Nilesh Suriyarachchi[1], Rien Quirynen[2], John S. Baras[1], and Stefano Di Cairano[2]

Abstract— **Coordinating the flow of traffic through urban areas with multiple intersections is a complex problem whose solution has the potential to improve safety, increase throughput, and optimize energy efficiency. In addition to controlling traffic lights, the introduction of connected and automated vehicles (CAVs) offers opportunities in terms of additional sensing and actuation points within the traffic network. This paper proposes a centralized and a decentralized implementation for the joint coordination and control of both traffic signals and mixed traffic, including CAVs and human driven vehicles (HDVs), in a network of multiple connected traffic intersections. Mixed-integer linear programming (MILP) is used to compute safe control trajectories for both CAVs and traffic light signals, which minimize overall congestion and fuel consumption. Our approaches are validated using extensive traffic simulations on the SUMO platform and they are shown to provide improvements of around** 32**-**60%**,** 90**-**96% **and** 40**-**60% **in travel time, waiting time and fuel consumption, respectively, when compared to gap-based adaptive and timed traffic lights.**

I. INTRODUCTION

Coordination and control of vehicles travelling through urban areas is an open challenge in transportation networks. Most traditional control systems are based on timed traffic light signals or density-adaptive traffic light signals [1]. However, these systems may change with the introduction of connected automated vehicles (CAVs) that would allow far more sensing and actuation points within the traffic flow [2]. In addition to collecting traffic information via on-board sensors on CAVs, traffic control systems can suggest CAV actions that optimize the traffic flowing through an urban network [3]. Furthermore, while human driven vehicles (HDVs) cannot be directly controlled by these control systems, the presence of neighbouring CAVs and traffic lights significantly improves the degree of controllability of the overall traffic flow in the network.

The main objective of an urban traffic control system is to allow vehicles to travel through the network in the shortest possible time while minimizing overall fuel consumption. To achieve this goal, a central traffic controller (CTC) is responsible for generating both CAV and traffic light control trajectories while ensuring safety, fairness and optimal traffic throughput. Achieving these goals in the presence of mixed traffic adds an extra layer of complexity due to the need for sufficiently accurate predictions of HDV behavior.

Intelligent traffic light control has been studied for decades, with methods ranging from time-based switching to density-based switching using heuristics [4], optimization [5] and deep learning [6]. For multi-intersections, an adaptive dynamic programming approach was presented in [7]. These methods typically allow for reactive control which can be sub-optimal and lead to vehicles having to slow down to a stop at intersections. Additionally, most of existing research focuses on macro-level traffic modeling which often abstracts details and results in a loss of efficiency at the micro-level. Research interest is also placed in CAV control for navigating through unsignalized intersections [8], [9] and merging points [3], [10], [11]. However, these algorithms typically require all vehicles to be CAVs and deal with only local control for a single traffic intersection.

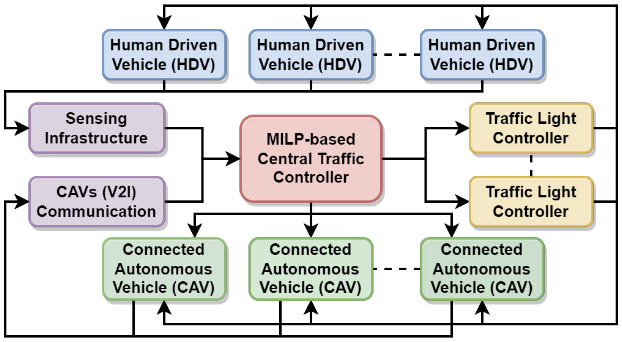
Human Driven Vehicle (HDV)
Human Driven Vehicle (HDV)
Human Driven Vehicle (HDV)
Sensing Infrastructure
CAVs (V2I) Communication
MILP-based Central Traffic Controller
Traffic Light Controller
Traffic Light Controller
Connected Autonomous Vehicle (CAV)
Connected Autonomous Vehicle (CAV)
Connected Autonomous Vehicle (CAV)

Fig. 1. Schematic of the proposed central traffic controller (CTC) directly controlling connected automated vehicles (CAVs) and indirectly affecting motion of human driven vehicles (HDVs) by commanding traffic lights.

In the context of centralized control of multi-intersection environments with mixed traffic including both CAVs and HDVs, receding horizon methods based on mixed-integer linear programming (MILP) using either a road segment-based [12] or time-based formulation [13] have recently been proposed. Many remaining challenges exist in extending this work to real-time capable systems that can operate under realistic conditions and high density traffic.

Contributions: We propose both centralized and decentralized MILP-based strategies for the joint computation of CAV trajectories and traffic light switching commands, targeting multi-intersection mixed traffic coordination, and we validate their performance in realistic traffic simulations. Compared to the prior work in [12], [13], our novel contributions are,

- We describe an MILP formulation for joint vehicle and intersection coordination, with support for multiple intersections, mixed traffic, lane changes, HDV predictions, and an advanced traffic light switching model.
- We propose a decentralized implementation by decoupling the MILP, resulting in multiple intersection traffic

[1]Electrical and Computer Engineering Department, University of Maryland, College Park, MD, USA. {nileshs,baras}@umd.edu
[2]Mitsubishi Electric Research Laboratories, Cambridge, MA, USA. {quirynen,dicairano}@merl.com

979-8-3503-2806-6/$31.00 ©2023 AACC

controllers (ITCs), to achieve real-time computations.

- We perform extensive validations of the proposed approach based on SUMO [14], using realistic simulations of vehicle demand, behavior and interactions.

In Section II, we describe the problem statement, modeling and objectives, and Section III describes the MILP for centralized coordination. The decentralized approach is presented in Section IV, and Section V presents the SUMO simulation results. Finally, Section VI concludes the paper.

Notation: $\mathbb{R}$, $\mathbb{R}_+$, $\mathbb{R}_{0+}$ ($\mathbb{Z}$, $\mathbb{Z}_+$, $\mathbb{Z}_{0+}$) are the set of real, positive real and nonnegative real (integer) numbers, and $\mathbb{B} = \{0, 1\}$. The logical operators *and*, *or*, *xor*, *not* are $\wedge$, $\vee$, $\veebar$, $\neg$, and the logical operators *implies* and *equivalent (if and only if)* are $\implies$, $\iff$. Inequalities between vectors are intended componentwise.

II. PROBLEM DESCRIPTION

In the multi-intersection traffic coordination problem, the central traffic controller (CTC) computes the most suitable CAV trajectories and traffic light state trajectories to allow for smooth traffic flow, as described in Figure 1. The presence of mixed traffic creates the need for accurate HDV behavior prediction in order to achieve the best performance. The overarching goals of the system involve the maximization of traffic throughput, while minimizing a combination of waiting time and fuel consumption. Here, we aim at executing the control center at a sampling frequency of 1 Hz, i.e., using a sampling time period $T_\mathrm{s} = 1$ sec, similar to [12], [13].

A. Modeling of Physical Road Network

In this work, we consider a case study based on a traffic network with 5 intersections, with connecting road segments as shown in Figure 2, each with 3 lanes in both directions. The distance between the centers of adjacent intersections is set to $100m$. We further use the index set $\mathcal{J} = \{1, \ldots, 5\}$ to denote each of the traffic intersections. The speed limit $\bar{v}$ of the road network is set to $15m/s$. The vehicles can be either CAVs or HDVs resulting in mixed traffic, and the CAV penetration level r_cav (ratio of CAVs to total number of vehicles) in the simulation can be adjusted. We use the index sets $\mathcal{V}^\mathrm{c}$ and $\mathcal{V}^\mathrm{h}$ to denote each of the CAVs and HDVs in the road network, where $\mathcal{V} = \mathcal{V}^\mathrm{c} \cup \mathcal{V}^\mathrm{h}$ and $\mathcal{V}^\mathrm{c} \cap \mathcal{V}^\mathrm{h} = \emptyset$.

B. Modeling of Controlled Vehicles (CAVs)

Each CAV $i \in \mathcal{V}^\mathrm{c}$ has an on-board control architecture, capable of computing actuation commands (throttle and braking) in order to follow a sequence of velocity targets set by the CTC. This on-board controller also computes the required steering angles to keep the vehicle in the lane and to execute lane change maneuvers as instructed by the CTC. From the CTC's perspective, CAVs can therefore be modeled as agents with known dimensions moving along the center of the lane. CAVs are also assumed to be able to relay information to and from intersection infrastructure using vehicle-to-infrastructure (V2I) communication. In this work, we assume perfect communication and do not model V2I network conditions such as delays and packet loss.

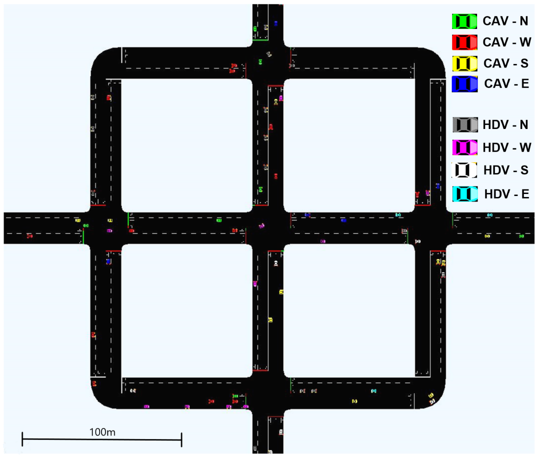

Fig. 2. Traffic road network with 5 intersections, where each road segment consists of 3 lanes in both directions, and 4 road segments where vehicles can flow in and out of the road network.

C. Modeling of Human Driven Vehicles (HDVs)

Similar to the work in [13], we model the predictions of HDV behavior using a switched dynamical system to represent reactions of HDVs to the, possibly changing, traffic rules. In the present work, we model each HDV $i \in \mathcal{V}^\mathrm{h}$ using the following state-dependent switched dynamics:

- HDV stops at traffic light, **if** traffic light for its particular traffic direction is red and HDV is within a predetermined distance from stopping zone, **otherwise**
- HDV follows leading vehicle and maintains safe following distance, **if** leading vehicle is within a particular distance in front of HDV, **otherwise**
- HDV travels at a fixed desired target speed.

In Section III, we describe how these switched dynamics can be modeled using mixed-integer constraints.

D. Traffic Light Intersection Model

Traffic lights at intersections are modeled as a collection of finite states where a state contains the status of each of the individual light signals of a particular intersection. The traffic lights of a junction allow vehicles to pass through the intersection $j \in \mathcal{J}$ in a specific direction $d \in \mathcal{D}$ if the traffic light phase status $\psi_j^d = 1$ (green), and no vehicles can pass through the intersection j in the direction d if $\psi_j^d = 0$ (red). For a 4-way intersection, where each road segment contains 3 lanes, there are 12 possible directions as shown in Figure 3, i.e., we define the index set $\mathcal{D} = \{1, \ldots, 12\}$.

A state Φ_j^s of intersection $j \in \mathcal{J}$ is defined as a collection of non-conflicting traffic directions that are allowed to enter the intersection j simultaneously. For the 4-way intersection, with 3 lanes in each road segment, 9 possible states can be identified, i.e., we define the index set $\mathcal{S} = \{1, \ldots, 9\}$. A mapping from state Φ_j^s to direction ψ_j^d is introduced in Table I for each intersection $j \in \mathcal{J}$, state $s \in \mathcal{S}$, and direction $d \in \mathcal{D}$. Here, $\psi_j^d = 0$ or $\psi_j^d = 1$ indicates that a specific direction's traffic light is red or green, respectively.

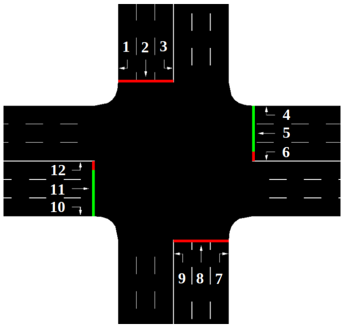

Fig. 3. Each 4-way traffic intersection, with 3 lanes in each road segment, includes traffic lights to control the flow in each of 12 possible directions.

State (Φ)	Phase Direction (ψ)											
	1	2	3	4	5	6	7	8	9	10	11	12
1	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	1	1	0	0	0	0	1	1	0
3	1	0	0	0	0	1	1	0	0	0	0	1
4	1	1	0	0	0	0	1	1	0	0	0	0
5	0	0	1	1	0	0	0	0	1	1	0	0
6	1	0	0	0	0	0	0	0	0	1	1	1
7	0	0	0	1	1	1	1	0	0	0	0	0
8	0	0	0	0	0	0	1	1	1	1	0	0
9	1	1	1	1	0	0	0	0	0	0	0	0

TABLE I

MAPPING FROM 9 TRAFFIC LIGHT STATES TO THE VALUE 0 (RED) OR 1 (GREEN) FOR EACH OF THE 12 TRAFFIC LIGHT DIRECTIONS.

III. MILP FORMULATION FOR CENTRALIZED TRAFFIC COORDINATION

The centralized traffic controller (CTC) is responsible for computing control trajectories for CAVs, including target velocities and lane changes, and traffic light commands while enforcing traffic rules and predicting the behavior of HDVs. Compared to the prior work in [13], the MILP-based CTC formulation in the present paper includes lane-change behavior for both CAVs and HDVs, as well as more advanced modeling for traffic lights and HDV predictions.

A. CAV Control Trajectories

The vehicle dynamics involve the evolution of its position state p_i along the direction of motion in the traffic network (see Fig. 2), according to the target velocity v_i, i.e.,

$$p_i(t+1) = p_i(t) + \frac{v_i(t) + v_i(t+1)}{2}\Delta t, \tag{1}$$

for each CAV $i \in \mathcal{V}^{\mathrm{c}}$, and where Δt is the discretization time. The initial target velocity $v_i(0)$ is set to the current vehicle's velocity, given by sensors and/or through V2I communication. To ensure realistic target velocities, the velocity assigned by the CTC to CAVs needs to be within the speed limit, $v_i(t) \leq \bar{v}_i$, and the maximum acceleration $\bar{a}_i^{\max}$ and deceleration $\bar{a}_i^{\min}$ limit of each vehicle

$$v_i(t) - \bar{a}_i^{\min}\Delta t \leq v_i(t+1) \leq v_i(t) + \bar{a}_i^{\max}\Delta t. \tag{2}$$

In order to model the lateral dynamics of vehicles, we introduce binary variables, $\sigma_i^h = 1$ if vehicle $i \in \mathcal{V}$ is in lane h, where $h \in \{1, \ldots, m_{\mathrm{l}}\} = \mathbb{Z}_1^{m_{\mathrm{l}}}$ and m_{l} is the maximum number of lanes on a road segment. These variables model lane changes, ensure vehicles enter intersections in the correct lane and perform collision avoidance in multi-lane scenarios. At every time step, each vehicle is always in exactly one of the traffic lanes, i.e., $\sum_h \sigma_i^h = 1$.

Lane change behavior: Lateral CAV dynamics are modeled as discrete switching between lanes. We introduce two binary variables $l_{i,\mathrm{u}}^{\mathrm{c}}(t), l_{i,\mathrm{d}}^{\mathrm{c}}(t) \in \{0,1\}$ to represent a lane change up and down respectively as follows,

$$\begin{aligned} \sigma_i^h(t) \ \wedge \ l_{i,\mathrm{u}}^{\mathrm{c}}(t) &\implies \sigma_i^{h+1}(t+1), \\ \sigma_i^h(t) \ \wedge \ l_{i,\mathrm{d}}^{\mathrm{c}}(t) &\implies \sigma_i^{h-1}(t+1). \end{aligned} \tag{3}$$

The implications in (3) can be enforced using the following inequality constraints for $i \in \mathcal{V}$ and $t \in \mathbb{Z}_0^{N-1}$

$$\begin{aligned} 1 - (1 - \sigma_i^h(t)) - (1 - l_{i,\mathrm{u}}^{\mathrm{c}}(t)) \leq \sigma_i^{h+1}(t+1) \leq 1, \\ 1 - (1 - \sigma_i^h(t)) - (1 - l_{i,\mathrm{d}}^{\mathrm{c}}(t)) \leq \sigma_i^{h-1}(t+1) \leq 1. \end{aligned} \tag{4}$$

Similarly, we implement the implication for no lane change

$$\sigma_i^h(t) \ \wedge \ \neg l_{i,\mathrm{u}}^{\mathrm{c}}(t) \ \wedge \ \neg l_{i,\mathrm{d}}^{\mathrm{c}}(t) \implies \sigma_i^h(t+1). \tag{5}$$

Finally, we prevent lane changes both up and down at the same time, along with lane change down from lowest lane, and lane change up from the highest lane as

$$\begin{aligned} &l_{i,\mathrm{u}}^{\mathrm{c}}(t) + l_{i,\mathrm{d}}^{\mathrm{c}}(t) \leq 1, \\ &l_{i,\mathrm{d}}^{\mathrm{c}}(t) + \sigma_i^1(t) \leq 1, \quad l_{i,\mathrm{u}}^{\mathrm{c}}(t) + \sigma_i^{m_{\mathrm{l}}}(t) \leq 1. \end{aligned} \tag{6}$$

B. Collision Avoidance

Collision avoidance constraints are placed between vehicle pairs whose trajectories may lead to potential collisions. Specifically, vehicles within the same road segments and adjoining road segments are paired up for collision checking. For these vehicle pairs, we enforce that the lead vehicle should remain ahead of the following vehicle, while both vehicles are in the same lane,

$$\begin{aligned} &\{p_{i_1}(t) + R_{i_1,i_2} \leq p_{i_2}(t) + L_{i_1} + d_{i_2}^{\mathrm{s}}(t) - s_{i_2}^{\mathrm{c}}\} \\ &\qquad\qquad \wedge \ \{\sigma_{i_1}^h(t+1) = \sigma_{i_2}^h(t+1)\} \\ &\quad \implies p_{i_1}(t+1) + R_{i_1,i_2} + L_{i_2} + d_{i_1}^{\mathrm{s}}(t) \leq p_{i_2}(t+1) + s_{i_1}^{\mathrm{c}}, \\ &\{p_{i_1}(t+1) + R_{i_1,i_2} \leq p_{i_2}(t+1) + L_{i_1} + d_{i_2}^{\mathrm{s}}(t) - s_{i_2}^{\mathrm{c}}\} \\ &\qquad\qquad \wedge \ \{\sigma_{i_1}^h(t+1) = \sigma_{i_2}^h(t+1)\} \\ &\quad \implies p_{i_1}(t) + R_{i_1,i_2} + L_{i_2} + d_{i_1}^{\mathrm{s}}(t) \leq p_{i_2}(t) + s_{i_1}^{\mathrm{c}}, \end{aligned} \tag{7}$$

for $h \in \mathbb{Z}_1^{m_{\mathrm{l}}}$, two vehicles $i_1, i_2 \in \mathcal{V}$, $t \in \mathbb{Z}_0^{N-1}$, and R_{i_1,i_2} denotes the initial relative distance between vehicles i_1 and i_2, and $s_i^{\mathrm{c}} \geq 0$ represents the slack variable that is added to ensure feasibility in the presence of model mismatch and disturbances. The collision avoidance constraints in (7) take into account spatial dimensions of vehicles and safety margins, i.e., including the vehicle lengths L_i and safety distances $d_i^{\mathrm{s}}(t) = D^{\mathrm{s}} + k^{\mathrm{v}} v_i(t)$ for each $i \in \mathcal{V}$, using a base value D^{s} and a velocity-dependent component with coefficient k^{v}. The presence of the slack variables s_i^{c}, in combination with the safety margins $d_i^{\mathrm{s}}(t)$, account for possible mismatch between predicted and actual vehicle behavior. In addition, the receding horizon implementation of the control center solves the MILP at each sampling time period $T_{\text{s}} = 1$ sec in order to correct possible prediction errors. Note that the constraints in (7) prevent lane changes that are predicted to lead to collisions at future time steps.

C. Intersection Membership

We introduce the matrix F_{sd} to model the mapping from traffic light states to individual traffic light signals for each direction, as shown in Table I. The relationship between traffic light states and individual traffic light signals for intersection $j \in \mathcal{J}$ is enforced by the constraints

$$\psi_j^d(t) \geq F_{sd}\,\Phi_j^s(t), \quad \psi_j^d(t) \leq 1 - (1 - F_{sd})\,\Phi_j^s(t), \tag{8}$$

for all states $s \in \mathcal{S}$ and all directions $d \in \mathcal{D}$ in each intersection $j \in \mathcal{J}$. Additionally, the traffic lights should be in exactly one state at any time $\sum_{s \in \mathcal{S}} \Phi_j^s = 1$.

To represent the presence of vehicle $i \in \mathcal{V}$ in intersection $j \in \mathcal{J}$ at time $t \in \mathbb{Z}_0^N$, we introduce binary variables $\delta_{ij}^d(t) \in \{0, 1\}$. We also introduce a given binary value $a_{ij}^d \in \{0, 1\}$ that indicates if a vehicle $i \in \mathcal{V}$ intends to cross intersection $j \in \mathcal{J}$ in direction $d \in \mathcal{D}$, as part of its future route plan. For each vehicle, we know the relative position at which it enters $P_{i,j}^{\text{in}}$ and exits $P_{i,j}^{\text{out}}$ the intersection. Therefore, we set the variables δ_{ij}^d based on the following bi-conditional

$$p_i(t) \in [P_{i,j}^{\text{in}} + s_i^\delta, P_{i,j}^{\text{out}} - s_i^\delta] \iff \sum_d a_{ij}^d \delta_{ij}^d(t) = 1, \tag{9}$$

where $s_i^\delta(t) \geq 0$ denotes a slack variable to ensure feasibility in the presence of model mismatch.

We can impose an upper bound on the number of vehicles that can be inside the intersection at the same time

$$\sum_i \delta_{ij}^d(t) \leq C_j^d\, \psi_j^d(t), \quad \forall j, d, \tag{10}$$

where C_j^d is the intersection capacity (upper bound based on physical intersection dimensions) for a specific direction $d \in \mathcal{D}$ and ψ_j^d is the traffic light signal for that direction, i.e., $\psi_j^d = 0$ (red) or $\psi_j^d = 1$ (green). Constraint (10) additionally ensures that vehicles can only enter the intersection if the corresponding traffic light signal is green.

D. HDV Modeling and Prediction

Using the HDV model in Section II-C, and depending on which mode is currently active for each HDV $i \in \mathcal{V}^{\text{h}}$, we compute a safe reference velocity $v_i^{\text{s}}(t)$ that is used to predict the HDV trajectories. To model mode switching, we introduce the binary variables $b_i^{\text{f}}(t), b_i^{\text{x}}(t) \in \{0, 1\}$ to indicate whether an HDV $i \in \mathcal{V}^{\text{h}}$ is in the lead vehicle following mode or stopping for red traffic light mode, respectively.

1) HDV lead vehicle following: For each HDV $i \in \mathcal{V}^{\text{h}}$, we introduce the index $i_1 \in \mathcal{V}$ to denote its lead vehicle, if a vehicle exists ahead of the HDV within a predetermined distance from the HDV. This lead vehicle could be either a CAV or another HDV. The safe following distance is defined using a base value D^{f} and a velocity-dependent component with coefficient k^{f}. In order to define the binary variable $b_i^{\text{f}}(t)$, we implement the following bi-conditional

$$p_{i_1}(t) + R_{i_1,i} - p_i(t) - L_{i_1} \leq D^{\text{f}} + k^{\text{f}} v_i(t) \iff b_i^{\text{f}}(t) = 1, \tag{11}$$

where $R_{i_1,i}$ denotes the initial relative distance between vehicles $i_1 \in \mathcal{V}$ and $i \in \mathcal{V}^{\text{h}}$, and given the length L_{i_1}.

2) HDV stopping at red traffic light: Setting the binary indicator variable $b_i^{\text{x}}(t)$ for stopping at a red light involves both the state of the associated traffic light signal and the set membership of the HDV in the braking region prior to the intersection. We introduce two binary variables b_i^{l} and b_i^{u} such that the HDV is in the braking region when both these variables are 1. For each HDV $i \in \mathcal{V}^{\text{h}}$, we denote the initial distance to the next intersection as P_i^{tl}, and the safe stopping distance is defined using a base value D^{I} and a velocity-dependent component with coefficient k^{I}. We implement the following implications

$$\begin{aligned} P_i^{\text{tl}} - p_i(t) \leq D^{\text{I}} + k^{\text{I}} v_i(t) &\implies b_i^{\text{l}}(t) = 1, \\ p_i(t) \leq P_i^{\text{tl}} &\implies b_i^{\text{u}}(t) = 1, \\ b_i^{\text{l}}(t) = 1 \wedge b_i^{\text{u}}(t) = 1 \wedge \psi_j^d(t) = 0 &\iff b_i^{\text{x}}(t) = 1, \end{aligned} \tag{12}$$

where $j \in \mathcal{J}$ denotes the next intersection such that the HDV stops if and only if the HDV is inside the braking region and the corresponding traffic light is red.

3) HDV safe reference velocity: The safe reference velocity v_i^{s} for HDV $i \in \mathcal{V}^{\text{h}}$ is defined based on the indicators b_i^{f} and b_i^{x}, using the following implications

$$\begin{aligned} b_i^{\text{x}}(t) = 1 &\implies v_i^{\text{s}}(t) = 0, \\ b_i^{\text{x}}(t) = 0 \wedge b_i^{\text{f}}(t) = 1 &\implies v_i^{\text{s}}(t) = v_{i_1}(t), \\ b_i^{\text{x}}(t) = 0 \wedge b_i^{\text{f}}(t) = 0 &\implies v_i^{\text{s}}(t) = \bar{v}, \end{aligned} \tag{13}$$

where v_{i_1} denotes the velocity of the leading vehicle $i_1 \in \mathcal{V}$ for HDV $i \in \mathcal{V}^{\text{h}}$, and $\bar{v}$ denotes the desired target speed.

4) HDV prediction model: The HDV prediction must respect the HDV's acceleration limits. We introduce the binary variables $b_i^{\max}$ and $b_i^{\min}$ to check whether these limits are reached. The HDV velocity prediction is given by

$$\begin{aligned} v_i^{\text{s}}(t+1) - v_i(t) \geq \bar{a}_i^{\max} \Delta t &\iff b_i^{\max}(t) = 1, \\ v_i^{\text{s}}(t+1) - v_i(t) \leq -\bar{a}_i^{\min} \Delta t &\iff b_i^{\min}(t) = 1, \end{aligned} \tag{14}$$

$$\begin{aligned} b_i^{\max}(t) = 1 &\implies v_i(t+1) = v_i(t) + \bar{a}_i^{\max} \Delta t, \\ b_i^{\min}(t) = 1 &\implies v_i(t+1) = v_i(t) - \bar{a}_i^{\min} \Delta t, \\ b_i^{\max}(t) = 0 \wedge b_i^{\min}(t) = 0 &\implies v_i(t+1) = v_i^{\text{s}}(t+1). \end{aligned} \tag{15}$$

E. Traffic Light Timing Constraints

We introduce a binary variable $l_{j,s}^{\text{tl}}(t) \in \{0, 1\}$ for each traffic light state $s \in \mathcal{S}$ of each intersection, where $l_{j,s}^{\text{tl}}(t) = 1$ if and only if the traffic light state changes, i.e.,

$$\Phi_j^s(t+1) - \Phi_j^s(t) \leq l_{j,s}^{\text{tl}}(t) \leq \Phi_j^s(t+1) + \Phi_j^s(t), \tag{16}$$

$$\Phi_j^s(t) - \Phi_j^s(t+1) \leq l_{j,s}^{\text{tl}}(t) \leq 2 - \Phi_j^s(t+1) - \Phi_j^s(t). \tag{17}$$

We introduce a variable $\hat{l}_j^{\text{tl}}(t) \in \{0, 1\}$ to denote if the overall state of the traffic intersection has changed, i.e.,

$$\sum_{s \in \mathcal{S}} l_{j,s}^{\text{tl}}(t) - 1 \leq \hat{l}_j^{\text{tl}}(t) \leq \sum_{s \in \mathcal{S}} l_{j,s}^{\text{tl}}(t). \tag{18}$$

Then, depending on the binary variable $\hat{l}_j^{\mathrm{tl}}(t)$, a timer variable $t_j^{\mathrm{tl}}(t)$ can be reset or updated as follows

$$-(1-\hat{l}_j^{\mathrm{tl}}(t))M \le t_j^{\mathrm{tl}}(t+1) \le (1-\hat{l}_j^{\mathrm{tl}}(t))M,$$
$$t_j^{\mathrm{tl}}(t)+\Delta t-\hat{l}_j^{\mathrm{tl}}(t)M \le t_j^{\mathrm{tl}}(t+1) \le t_j^{\mathrm{tl}}(t)+\Delta t+\hat{l}_j^{\mathrm{tl}}(t)M. \tag{19}$$

We additionally impose a minimum time between two consecutive traffic light change commands as well as an upper bound for the counter variable as follows

$$t_{\min}-M(1-\hat{l}_j^{\mathrm{tl}}(t)) \le t_j^{\mathrm{tl}}(t) \le t_{\max}. \tag{20}$$

F. MILP Objective Function

The objective function of the proposed MILP formulation for the CTC reads as the maximization of

$$J=\sum_{t=1}^{N}\sum_{i\in\mathcal{V}}\big(w_1\,p_i(t)-w_2\,|\sigma_i(t)-\sigma_i^{\mathrm{ref}}| \tag{21a}$$
$$-w_3\,l_{i,\mathrm{u}}^{\mathrm{c}}(t)-w_3\,l_{i,\mathrm{d}}^{\mathrm{c}}(t)-w_4\,s_i^{\delta}(t)-w_5\,s_i^{\mathrm{c}}(t)\big), \tag{21b}$$

with user-defined weight values $w_i \ge 0$ for $i=1,\ldots,5$. The first term is to maximize the traveled distance $p_i(t)$ of all vehicles in (21a). We penalize vehicles for not being in their preferred lane by minimizing $|\sigma_i(t)-\sigma_i^{\mathrm{ref}}|$, in which the absolute value can be handled by defining auxiliary variables in the MILP formulation. We introduced a penalty on unnecessary lane changes by minimizing $l_{i,\mathrm{u}}^{\mathrm{c}}(t), l_{i,\mathrm{d}}^{\mathrm{c}}(t)$ in (21b). In addition, we penalize the slack variables $s_i^{\delta}(t), s_i^{\mathrm{c}}(t) \ge 0$ that ensure feasibility in the presence of model mismatch and disturbances. Large penalty weight values $w_4, w_5 \gg 0$ are used to ensure that a solution with $s_i^{\delta}(t)=0$ and $s_i^{\mathrm{c}}(t)=0$ is computed, whenever feasible.

IV. DECENTRALIZED TRAFFIC COORDINATION

The main concern with the centralized control center is the combinatorial complexity of the MILP solution, i.e., the computational cost generally increases exponentially with the number of vehicles and the prediction horizon length. As shown in Section V, given our desired update frequency of 1 Hz, the centralized system is often not tractable under current hardware/processing limitations and with current state-of-the-art MILP solvers such as GUROBI [15]. A natural way to decompose the centralized coordination problem is by solving a subproblem for each traffic intersection individually with considerably fewer optimization variables.

A. Decentralized Architecture

In the proposed decentralized implementation, each intersection is assumed to have an independently operating control center, which is further referred to as an intersection traffic controller (ITC). The ITC at each intersection operates fundamentally similar to the centralized control center, based on the same MILP formulation as described in Section III, but with each of the variables restricted to only the relevant intersection, its traffic lights and immediately approaching vehicles within a predetermined *control zone* around the traffic intersection. Each ITC computes CAV velocities and lane change trajectories as well as the traffic light phase switching trajectories for its particular intersection, and the computations for each ITC may be executed in parallel on separate computing units. Adapting the MILP formulation from Section III to the decentralized implementation, for each ITC, we consider one intersection $j \in \mathcal{J}$ and a subset $\mathcal{V}_j \subseteq \mathcal{V}$ of the vehicles in the traffic network.

B. Vehicle Assignment to MILP-ITCs

In the decentralized implementation, the HDV modeling described in Section III-D can be used to model both HDVs and CAVs that are controlled by a different ITC. Therefore, each CAV could be assigned to a particular ITC "for control" or "for prediction". This is used in the proposed decentralized implementation to increase safety, especially when vehicles transition from one ITC to the next. The assignment of vehicles to the subsets $\mathcal{V}_j$, for each ITC $j \in \mathcal{J}$, is performed according to the following rules:

1) Each HDV is assigned to an intersection's ITC for prediction, if it is within a predetermined distance.
2) Each CAV is assigned to their next intersection's ITC for control, if it is within a predetermined distance.
3) Each CAV is assigned to their previous intersection's ITC for prediction, i.e., it will be treated as HDV by that MILP-ITC, if it is within a predetermined distance.

The latter set of rules ensure that the velocity and lane change commands for each CAV are computed by at most one ITC. If a CAV is not assigned to any ITC, given the predetermined threshold distance, then control is returned to the on-board control architecture. In the region immediately after a traffic intersection, both HDVs and CAVs are treated equally and their behavior is predicted to prevent possible collisions with any vehicles further upstream. Each ITC provides controls to CAVs only prior to the intersection and while the vehicle is physically within the intersection. Control is handed back to the vehicle or to the next junction's ITC (if junctions are very close) after the CAVs exit the intersection.

V. NUMERICAL VALIDATION RESULTS WITH SUMO SIMULATOR

In this Section, we present extensive simulation results to validate the performance of the proposed implementations for the traffic coordination and control system, and to investigate its computational tractability. The mixed traffic simulations are built on the SUMO [14] platform. The control systems are implemented in Matlab and the MILP problems are solved using GUROBI [15]. The communication between Matlab and SUMO is handled by the traci4matlab [16] software. The simulations are executed on a desktop with an Intel i7-10700k processor and 64GB of RAM.

A. Traffic Simulation

The traffic inflow from each of the four input directions (see Fig. 2) are kept equal to each other, resulting in a balanced flow of vehicles in the road network. The rate of traffic inflow for CAVs and HDVs is varied independently, in order to create multiple scenarios with varying traffic inflow rates r_{in} and CAV penetration levels r_{cav} (ratio of CAVs to total number of vehicles). Vehicles enter the simulation according to a Poisson process with expected value of arrivals equal to the desired input flow rate for each inflow road. Each vehicle entering the network has an assigned destination road and the system chooses between possible routes from source to destination based on predefined probabilities to provide balanced traffic flow throughout the network. In the simulated vehicle model of `SUMO` [14], parameters for the level of driver imperfection are set to low for CAVs and relatively high for HDVs. This results in lower predictability of HDV behavior due to higher variance in human actions.

B. Comparison Metrics and Baselines

The key metrics used in evaluating comparable traffic control systems are the average travel time of vehicles flowing through the network, the average time spent waiting at traffic lights per vehicle and the average fuel consumption per vehicle. In addition to these metrics, we also present the average computation times (τ) for each of the proposed control architectures. The runtime (τ_{ITC}) of the decentralized architecture involves solving multiple MILPs in parallel, one for each intersection's ITC. We report both the average over time of the CPU runtime for the most expensive MILP at each time step (dark blue bar in Figures 4,5,6), as well as the average over time of cumulative CPU runtimes to solve all MILPs at each time step (light blue bar in Figures 4,5,6),

$$\tau_{\text{ITC}}^{\max} = \frac{1}{T}\sum_{t=1}^{T}\max_{j\in\mathcal{J}}\tau_{\text{ITC}}^{j}(t), \quad \tau_{\text{ITC}}^{\text{sum}} = \frac{1}{T}\sum_{t=1}^{T}\sum_{j\in\mathcal{J}}\tau_{\text{ITC}}^{j}(t). \tag{22}$$

We use the standard baseline of a time-based traffic lights switching system as a reference. The traffic light timings are tuned to suit the densities being tested. All four inflow roads in an intersection are treated equally in terms of traffic light timing since the overall inflow to the network is balanced. Furthermore, as a more competitive reference, we implemented a gap-based adaptive traffic light control strategy in `SUMO` [14]. This adaptive strategy allows traffic lights to dynamically adapt to current traffic conditions in the incoming lanes of an intersection by prolonging traffic phases whenever a continuous stream of traffic is detected.

C. Performance Analysis

The overall performance of the systems show that both centralized and decentralized control strategies significantly outperform the timed traffic lights. We observed considerable improvements of around 60%, 96% and 60% in travel time, waiting time and fuel consumption, respectively, when using the centralized strategy over the timed traffic lights method. Similarly, when comparing the centralized strategy against the adaptive traffic lights method, we observed improvements of around 32%, 90% and 40% in travel time, waiting time and fuel consumption, respectively. These significant reductions in waiting times correspond to reduced idling of vehicles at traffic lights, which leads to more eco-driving strategies, which in turn results in fuel savings.

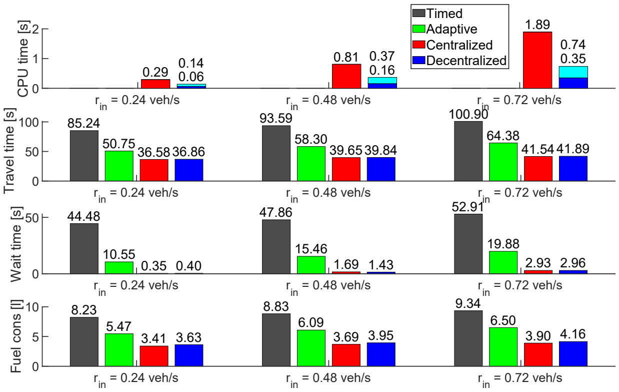

Fig. 4. `SUMO` results over 2000 s to compare centralized MILP, decentralized MILPs versus timed and adaptive traffic lights for increasing inflow rates $r_{\text{in}} = 0.24, 0.48$ and 0.72 veh/s, with $N = 10, r_{\text{cav}} = 50\%$.

The centralized strategy also outperforms the decentralized strategy in most metrics by a smaller margin, at the cost of increased computation times. We observed a reduction in performance of around 2%, 3% and 9% in travel time, waiting time and fuel consumption, respectively, when using the decentralized strategy over the centralized strategy. However, the decentralized strategy CPU computation times are around 89% less than the centralized strategy.

1) Impact of vehicle inflow rate: As the inflow rate r_{in} increases, as observed in Figure 4, the travel time, waiting time and fuel consumption show worse performance across all the control strategies. High inflow rates cause the timed traffic lights to have 100 sec average travel time and 53 sec average waiting time, but the CTC reduces these values to 42 sec and 3 sec, respectively. Even compared to the adaptive traffic lights, the CTC demonstrates a 36%, 85% and 40% improvement in travel time, waiting time and fuel consumption. The decentralized ITCs allow for fast computations below 1 sec, while only resulting in around a 5% suboptimality in terms of other performance metrics.

2) Impact of CAV penetration: The percentage of CAVs play a major role in the performance of the control system. A greater amount of CAVs offers multiple actuation points in the traffic flow which leads to better controllability of the overall traffic network. For example, as observed in Figure 5, using a $N = 10$ planning horizon and 0.6 veh/sec inflow rate, in the CTC we observe performance gains of 8%, 46% and 24% in travel time, waiting time and fuel consumption, respectively, when switching from 20% to 80% CAV penetration. Furthermore, we also observe that prediction of HDVs is a more computationally expensive process than the CAV trajectory computation, due to the increased number of integer variables involved. Thereby, the average computation times reduce by 36%, when switching from 20% to 80% CAV penetration.

3) Impact of planning horizon: We observe that longer planning horizons lead to improved performance in terms of the key metrics but also cause the computation time to increase rapidly. As shown in Figure 6, at $N = 5$ horizon

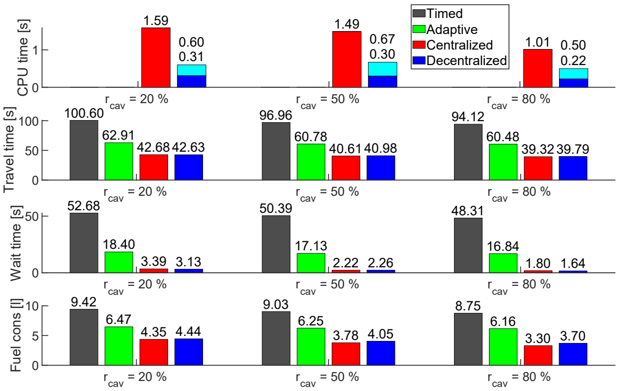

Fig. 5. SUMO results over 2000 s to compare centralized MILP, decentralized MILPs versus timed and adaptive traffic lights for increasing CAV penetration levels $r_{\text{cav}} = 20, 50$ and 80%, with $N = 10, r_{\text{in}} = 0.60$ veh/s.

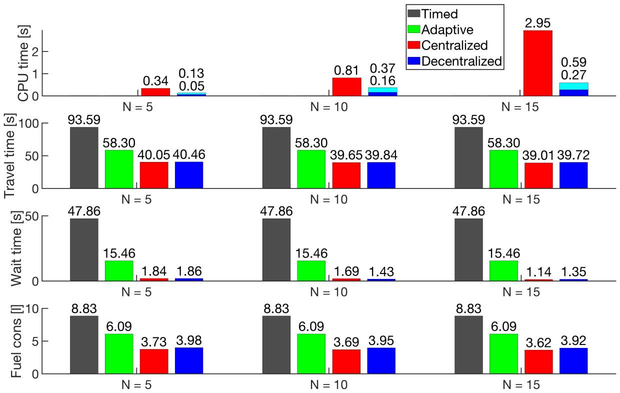

Fig. 6. SUMO results over 2000 s to compare centralized MILP, decentralized MILPs versus timed and adaptive traffic lights for increasing horizon lengths $N = 5, 10, 15$, with $r_{\text{in}} = 0.48$ veh/s, $r_{\text{cav}} = 50\%$.

length, the CTC is capable of computing on average in about 0.3 sec. When the horizon length is increased to $N = 15$, the average computation time increases significantly to around 3 sec. This is where the decentralized strategy with multiple ITCs becomes more suitable. ITCs offer similar performance improvements obtained by increasing horizon length while keeping total computation times under 0.5 sec, even at the $N = 15$ horizon length.

4) Real-time feasibility: To achieve real-time feasibility at a 1 Hz control frequency, the CTC/ITC needs to solve its MILP problem in less than 1 sec. The CTC is able to achieve this threshold only at low traffic inflow rates and shorter planning horizons. However, at higher inflow rates and planning horizons, the CTC's average computation time can easily exceed 10 sec. The average runtime of the ITCs remain below 0.8 sec, making the decentralized strategy a more practically implementable real-time system under current hardware limitations.

VI. CONCLUSIONS

We presented an MILP approach for the computation of control trajectories for CAVs and traffic light signals in a network of multiple connected intersections with mixed traffic, including both CAVs and HDVs. Using extensive SUMO traffic simulations, our proposed centralized traffic controller demonstrated a considerable increase in performance, with around 90-96%, 32-60%, and 40-60% improvements in wait times, travel times and fuel consumption, respectively, over both adaptive and timed traffic light control methods. We also introduced a decentralized implementation, with multiple decoupled MILPs, which allows the system to be computationally tractable for higher traffic densities and longer planning horizons, while only causing a relatively small suboptimality in performance. Future work may involve reducing the optimality gap in the decentralized method by providing global information on traffic conditions.

REFERENCES

[1] M. Eom and B.-I. Kim, "The traffic signal control problem for intersections: a review," *European Transport Research Review*, vol. 12, no. 1, p. 50, Sep 2020.

[2] T. Ersal, I. Kolmanovsky, N. Masoud, N. Ozay, J. Scruggs, R. Vasudevan, and G. Orosz, "Connected and automated road vehicles: state of the art and future challenges," *Vehicle system dynamics*, vol. 58, no. 5, pp. 672–704, 2020.

[3] J. Rios-Torres and A. A. Malikopoulos, "A survey on the coordination of connected and automated vehicles at intersections and merging at highway on-ramps," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 5, pp. 1066–1077, 2017.

[4] A. Kanungo, A. Sharma, and C. Singla, "Smart traffic lights switching and traffic density calculation using video processing," in *Recent Advances in Eng. and Computat. Sciences (RAECS)*, 2014, pp. 1–6.

[5] B. De Schutter and B. De Moor, "Optimal traffic light control for a single intersection," *Europ. J. Control*, vol. 4, no. 3, 1998.

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

[7] D. Liu, W. Yu, S. Baldi, J. Cao, and W. Huang, "A switching-based adaptive dynamic programming method to optimal traffic signaling," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 50, no. 11, pp. 4160–4170, 2020.

[8] A. A. Malikopoulos and L. Zhao, "Optimal path planning for connected and automated vehicles at urban intersections," in *IEEE Conference on Decision and Control (CDC)*, 2019, pp. 1261–1266.

[9] N. Suriyarachchi, R. Chandra, J. S. Baras, and D. Manocha, "Gameopt: Optimal real-time multi-agent planning and control for dynamic intersections," in *2022 IEEE International Intelligent Transportation Systems Conference (ITSC)*, 2022.

[10] F. Ye, J. Guo, K. J. Kim, P. V. Orlik, H. Ahn, S. Di Cairano, and M. J. Barth, "Bi-level optimal edge computing model for on-ramp merging in connected vehicle environment," in *Proc. IEEE Intell. Veh. Symp.*, 2019, pp. 2005–2011.

[11] N. Suriyarachchi, F. M. Tariq, C. Mavridis, and J. S. Baras, "Real-time priority-based cooperative highway merging for heterogeneous autonomous traffic," in *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, 2021, pp. 2019–2026.

[12] S. Ravikumar, R. Quirynen, A. Bhagat, E. Zeino, and S. Di Cairano, "Mixed-integer programming for centralized coordination of connected and automated vehicles in dynamic environment," in *Proc. IEEE Conf. Control Technology and Applications*, 2021.

[13] R. Firoozi, R. Quirynen, and S. Di Cairano, "Coordination of autonomous vehicles and dynamic traffic rules in mixed automated/manual traffic," in *2022 American Control Conference (ACC)*, 2022, pp. 1030–1035.

[14] P. A. Lopez, M. Behrisch, L. Bieker-Walz, J. Erdmann, Y.-P. Flötteröd, R. Hilbrich, L. Lücken, J. Rummel, P. Wagner, and E. Wießner, "Microscopic traffic simulation using sumo," in *IEEE Int. Conf. on Intell. Transp. Systems*. IEEE, November 2018, pp. 2575–2582.

[15] L. Gurobi Optimization, "Gurobi optimizer reference manual," 2021.

[16] A. F. Acosta, J. E. Espinosa, and J. Espinosa, "TraCI4matlab: Enabling the Integration of the SUMO Road Traffic Simulator and Matlab® Through a Software Re-engineering Process," in *Modeling Mobility with Open Data*. Springer, 2015, pp. 155–170.