



A hybrid stochastic model predictive design approach for cooperative adaptive cruise control in connected vehicle applications

Sahand Mosharafian, Javad Mohammadpour Velni *

School of Electrical and Computer Engineering, University of Georgia, GA 30602, United States

ARTICLE INFO

Keywords:

Connected and automated vehicles
Cooperative adaptive cruise control
Model predictive control
Discrete hybrid stochastic automata

ABSTRACT

Wireless communication among connected and automated vehicles (CAVs) enables cooperative driving, particularly using cooperative adaptive cruise control, hence allowing CAVs to safely move closer to each other and improving traffic flow. However, reducing the gap between vehicles may cause collision, especially when a sudden deceleration in the vehicle platoon occurs. Furthermore, if some of the CAVs in the platoon lose communication, even for a short period, they may detect sudden changes in traffic with delay, and their reaction to avoid collision could be unsuccessful. Hence, an efficient emergency braking system can help preserve safety. This becomes even more important in high-speed driving, where a crash can put human lives in danger. This paper presents a discrete hybrid stochastic model predictive control approach to achieve a safe and efficient traffic system through CAV platooning. Three operational modes for vehicles are considered: free following, warning, and emergency braking. Each CAV in the free-following mode basically follows its preceding vehicle while enforcing maximum allowable deceleration in the emergency-braking mode. In the warning mode, the vehicle aims to slightly increase its headway and velocity difference from its predecessor to avoid possible danger. The warning/emergency-braking mode may be activated when the speed difference between a CAV and its preceding vehicle drops below a threshold. It is further assumed that vehicle distance sensing is subject to error. Each vehicle shares its current location, velocity, and future acceleration profile, calculated by solving a mixed-integer programming problem, with its follower vehicles. Simulation studies demonstrate the efficacy of the proposed control design approach compared to existing controllers in the literature in the presence of intermittent communication.

1. Introduction

Modern vehicles are equipped with different driver assistance systems to facilitate driving. These systems help drivers with doing repetitive tasks such as stop and start while improving safety and reducing fuel consumption (Moser, Waschl, Kirchsteiger, Schmied, & Del Re, 2015). Furthermore, advanced driver assistance systems may improve the traffic system by increasing the road capacity (Vahidi & Eskandarian, 2003). By increasing the popularity and demand for autonomous vehicles, vehicle platooning in highways and roads would be a way to increase the efficiency of the traffic system.

The main goal in the longitudinal vehicle platoon control is to enhance performance while preserving safety. Wireless communication among vehicles enables cooperative adaptive cruise control (CACC) which allows vehicles to safely move closer to their preceding vehicle. The CACC has the potential to improve the traffic system while its primary goal was to comfort driving experience (Van Arem, Van Driel, & Visser, 2006). Although CACC takes advantage of wireless communication, the communication is subject to failure. Hence, the controls

should be robust against data loss and communication failure (Ploeg, 2014). Connection failure resorts CACC to adaptive cruise control (ACC) if no action is taken to compensate for packet losses (Ploeg, Semsar-Kazerooni, Lijster, van de Wouw, & Nijmeijer, 2014).

Several robust CACC approaches have been proposed to overcome the impact of unexpected events on mobility and to preserve the safety and stability of the vehicle platoon (e.g., see Cui, Chen, Wang, Hu, & Park, 2021; Lan, Zhao, & Tian, 2020; Trudgen, Miller, & Velni, 2018; Van Nunen, Verhaegh, Silvas, Semsar-Kazerooni, & Van De Wouw, 2017). Furthermore, several research studies have focused on how to use the braking system during an emergency in a connected and automated vehicle (CAV) platoon. In Hasan, Balador, Girs, and Uhlemann (2019), a synchronized braking system for vehicles was introduced where a vehicle (e.g., the leader vehicle) does not immediately apply emergency braking; instead, it informs other vehicles through communication so that all vehicles perform the braking at the same time. This method relies on perfect communication, as otherwise, it may cause a collision. A procedure for adapting the deceleration rate according to the vehicle with the weakest braking capability was discussed

* Corresponding author.

E-mail address: javadm@uga.edu (J. Mohammadpour Velni).

in Murthy and Masrur (2016). In Turri, Besselink, and Johansson (2016) and Zhai, Liu, and Luo (2018), an MPC-based design is proposed, where preserving safety is considered as a constraint that should always be satisfied. In Zhai et al. (2018), there is a separate emergency braking controller which is activated if the original MPC optimization problem becomes infeasible. Neither one of the last two works considered an imperfect sensing system, and those methods are not suitable in the presence of intermittent communication. According to Wu, Lin, and Eskandarian (2019), three general solutions are available to cope with the safety problem during communication failure: (1) improving the communication protocol (e.g., see Lee & Chanson, 2002); (2) increasing the vehicle headway during failure (e.g., see Abou Harfouch, Yuan, & Baldi, 2017); (3) adapting CACC algorithms to cope with the safety problem (e.g., using a model predictive control approach). The method proposed in our paper to preserve safety is a fusion of first and third solutions.

Two general CACC approaches are available: state feedback control methods (e.g., Guo & Yue, 2014; Morbidi, Colaneri, & Stanger, 2013; Ploeg, 2014), and constrained-optimization methods which have more recently evolved to MPC (e.g., Kianfar, Falcone, & Fredriksson, 2015; Zhai et al., 2018; Zhou, Wang, & Ahn, 2019). State-feedback-based methods enable explicit system stability analysis while constrained-optimization methods allow considering multiple performance indices subject to hard constraints on the system dynamics and physical limitations of the vehicle (Zhou et al., 2019). Traditional MPC schemes are formulated in a centralized setting. However, due to the practical limitations for gathering data and the large size of the optimization problem, distributed MPC has emerged to address the limitations of centralized methods (Wang, Wu, & Barth, 2018).

This paper examines the CACC design problem with three operating modes for each vehicle: free following, warning, and emergency braking. In the free-following mode, each vehicle simply follows its preceding vehicle while the vehicle in the emergency-braking mode performs hard braking to avoid possible collision and preserve safety. Warning mode is considered to avoid the need to perform emergency braking as much as possible as the CAV in warning mode slightly increases its desired headway and reduces its desired velocity. While directly considering these modes in the controller design process penalizes unnecessary braking, it adds discrete variables to the underlying problem formulation. In order to formulate the system and design a controller that takes three modes into account, a hybrid model is needed. Since the vehicles distance measurements are subject to measurement error, a stochastic hybrid model is required to cope with that. Hence, discrete hybrid stochastic automata, introduced initially in Bemporad and Di Cairano (2010) suits the aforementioned stochastic system with both continuous and discrete variables. The distinctive feature of the proposed method is that each vehicle's optimal operating mode is chosen based on the predictive information of its predecessors. Hence, with a proper communication rate, vehicles are able to plan their acceleration profile to avoid emergency braking. In the case of communication losses, CACC may degrade to ACC for some vehicles in the platoon, in which case they would not have access to the predictive information for some time. If sudden deceleration occurs during this period, some vehicles may need to switch to warning or emergency braking mode.

The contributions of this work are as follows. The paper offers a new integrated perspective, that is, a discrete stochastic and optimal MPC design approach, for treating (and switching between) three operating modes, namely, warning, emergency braking and free following, in connected and automated vehicle applications. The proposed approach leads to a safer autonomous vehicle platoon than currently existing approaches. It is noted that with the proposed stochastic MPC-based design, vehicles would avoid unnecessary hard (emergency) braking — this is due to that the emergency braking is explicitly considered in the underlying optimization problem so that excessive braking is penalized. Furthermore, the proposed approach demonstrates strong

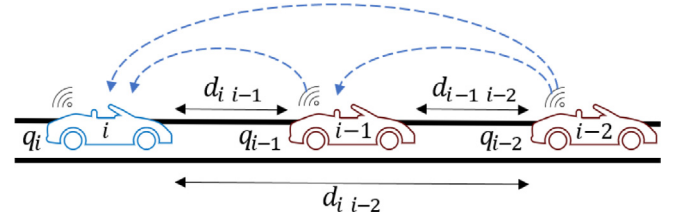


Fig. 1. A simple representation of the system model. The distance between vehicle i and vehicle j is represented by d_{ij} while q_i is the location of the rear bumper of the i th vehicle. The blue dashed lines show the information flow.

performance with a low vehicle-to-vehicle communication rate while also considering communication failure.

The remainder of the paper is organized as follows. In Section 2, the system model, as well as a brief introduction to discrete hybrid stochastic automata are presented. System model and operating constraints (DHSA) are presented in mixed logical dynamical (MLD) form in Section 3. The proposed model predictive control design method is discussed in Section 4. Simulation results and evaluation of the performance of the proposed controller are given in Section 5. Concluding remarks are finally made in Section 6.

2. Preliminaries

In this section, first the dynamic model of vehicles considered in this study are reviewed. Then, the framework in which the control design problem is formulated, i.e., the stochastic and hybrid setting, is described.

2.1. Description of the system model

A CACC system with N_v vehicles is considered in the paper. Notation $i \in \{0, 1, \dots, N_v - 1\}$ is used to represent the i th vehicle with $i = 0$ being the leader vehicle. As shown in Fig. 1, the distance between i th vehicle and j th vehicle where $j < i$, at time t is denoted by $d_{ij}(t)$ and defined as

$$d_{ij}(t) = q_j(t) - q_i(t) - l_i, \quad (1)$$

where q_i and q_j are the location of the i th and j th vehicles' rear bumper, respectively, and l_i is the length of the i th vehicle. In the sequel, it is assumed that $j < i$. Using a fixed time gap spacing policy, the desired spacing between vehicle i and its immediate predecessor is defined as

$$d_i^*(t) = h v_i(t) + r_i, \quad (2)$$

where $v_i(t)$ is the vehicle's speed at time t , h is the time gap, and r_i is the standstill distance. Based on (2), the desired distance between vehicles i and j can be calculated as

$$d_{ij}^*(t) = \sum_{k=j+1}^i d_k^*(t) + \sum_{k=j+1}^{i-1} l_k. \quad (3)$$

It is noted that, $d_{ij}^*(t)$ is equivalent to $d_i^*(t)$ if $j = i - 1$. Using (3), the spacing error between vehicles i and j is indicated as

$$\Delta d_{ij}(t) = d_{ij}(t) - d_{ij}^*(t). \quad (4)$$

Similarly, the speed difference between the i th and j th vehicle is considered as $\Delta v_{ij}(t) = v_j(t) - v_i(t)$ and therefore $\Delta \dot{d}_{ij}(t) = \Delta v_{ij}(t) - h \sum_{c=j+1}^i a_c(t)$ and $\Delta \dot{v}_{ij}(t) = a_j(t) - a_i(t)$, where $a_i(t)$ and $a_j(t)$ are the acceleration of the i th and j th vehicle at time t , respectively. Let us consider a linear model for the i th vehicle as

$$\dot{a}_i(t) = -\frac{1}{\tau_i} a_i(t) + \frac{1}{\tau_i} u_i(t), \quad (5)$$

where the time constant τ_i represents driveline dynamics. It is reasonable to assume that vehicles share their information with their

followers within a specific range. So, we assume that each vehicle receives information from its $m \geq 1$ front vehicles. The given system can be easily represented in the following state-space form (assuming that there exist m predecessor vehicles for i th vehicle):

$$\begin{aligned} \dot{x}_i(t) &= A_i x_i(t) + B_i u_i(t) + D_i a_i^m(t) \\ &= \begin{bmatrix} \mathbf{0}_{m \times m} & I_{m \times m} & -h * \mathbf{1}_{m \times 1} \\ \mathbf{0}_{m \times 2m} & -\mathbf{1}_{m \times 1} \\ \mathbf{0}_{1 \times 2m} & \frac{-1}{\tau_i} \end{bmatrix} x_i(t) \\ &\quad + \begin{bmatrix} \mathbf{0}_{2m \times 1} \\ \frac{1}{\tau_i} \end{bmatrix} u_i(t) + \begin{bmatrix} \mathcal{L}_{m \times m} \\ I_{m \times m} \\ \mathbf{0}_{1 \times m} \end{bmatrix} a_i^m(t), \end{aligned} \quad (6)$$

where $\mathbf{0}$ is a matrix with zero entries, I is the identity matrix, $\mathbf{1}$ is a matrix with one entries, and $\mathcal{L}$ is a lower diagonal matrix in which the diagonal entries are zero and the lower diagonal entries are $-h$. The vector of system states ($x_i(t)$) and predecessor vehicles' acceleration vector ($a_i^m(t)$) are sorted as follows

$$\begin{aligned} x_i(t) &= [\Delta d_{i-1}(t) \ \Delta d_{i-2}(t) \ \dots \ \Delta d_{i-m}(t) \\ &\quad \Delta v_{i-1}(t) \ \Delta v_{i-2}(t) \ \dots \ \Delta v_{i-m}(t) \ a_i(t)]^T, \\ a_i^m(t) &= [a_{i-1}(t) \ a_{i-2}(t) \ \dots \ a_{i-m}(t)]. \end{aligned}$$

If there are at least m predecessors for vehicle i ($i \geq m$), then A_i is $(2m+1) \times (2m+1)$, B_i is $(2m+1) \times 1$, and D_i is $(2m+1) \times m$; otherwise, m reduces to the number of the i th vehicle predecessors, which is i . For the leader vehicle ($i = 0$), $a_i^m(t)$ in (6) can be assumed to be zero.

Remark 1. It is assumed that each CAV measures its distance from its immediate predecessor, as well as the predecessor velocity while the distance measurements are subject to error. It is also assumed to have access to the other $m-1$ predecessors' location and speed through communication. Moreover, each CAV transmits its future acceleration profile to its m followers.

Using forward-time approximation for the first-order derivative, (6) can be written in discrete-time form. By considering measurement error ($n_i(k)$) in the formulation, the discrete-time state-space model for each follower is as follows

$$\begin{aligned} x_i(k+1) &= \\ &= (I + t_s A_i) x_i(k) + t_s B_i u_i(k) + t_s D_i a_i^m(k) + G n_i(k), \end{aligned} \quad (7)$$

where $G = [I \ \mathbf{0}_{1 \times 2m}]^T$, and t_s is the sampling time.

2.2. Introduction to discrete hybrid stochastic models

Following the notations initially presented in Bemporad and Di Cairano (2010), a stochastic system with both binary and continuous/discrete-time variables and inputs can be modeled using a discrete hybrid stochastic automata (DHSA) which consists of the following four components:

1. A switched affine model in the form of a linear difference equation as

$$x_c(k+1) = A_{i(k)} x_c(k) + B_{i(k)} u_c(k) + f_{i(k)}, \quad (8)$$

where $i(k)$ is the mode of the system, $x_c(k)$ is the vector of discrete-time (in general, can be continuous-time as well) states, and $u_c(k)$ is the input vector. In our CACC problem, the term f_i is built using the acceleration information of the predecessor vehicles as $f_i = t_s D_i a_i^m$.

2. An event generator which produces a binary output $\delta_e(k) = f_{EG}(x_c(k), u_c(k))$ such that

$$f_{EG}(x_c, u_c) = 1 \iff H_e x_c + J_e u_c + K_e \leq 0 \quad (9)$$

where H_e , J_e , and K_e are constant matrices representing state weights, input weights, and bias in linear event generator inequalities, respectively. These matrices are chosen based on the system properties, e.g., in our problem, they represent the thresholds for the CAV's operating modes. In other words, they define the threshold with which an event becomes active and are chosen based on the system's structure and definition of the events in the system (e.g., emergency braking event). Parameter δ_e is a binary variable that equals one (1) when the event generation constraint is satisfied. Defining binary variables enables transforming the system events into the mixed logical dynamical form. The main binary states in our problem include the vehicle's operating modes. Other auxiliary binary variables may be required so that the DHSA system can be represented in MLD form.

3. A mode selector that determines the operating mode of the system using

$$i(k) = f_{MS}(x_b(k), u_b(k), \delta_e(k)),$$

where f_{MS} is a Boolean function, and $x_b(k)$ and $u_b(k)$ are the vectors of binary states and binary inputs, respectively. It is noted that the mode of the system depends on the event generator's binary output δ_e .

4. A finite state machine (FSM) that includes the probabilistic part of the system and represents the stochastic transition from a binary state vector to another one and is described by

$$P[x_b(k+1) = \hat{x}_b] = f_{FSM}(x_b(k), u_b(k), \delta_e(k), \hat{x}_b),$$

where P denotes the probability. If $P[x_b(k+1)]$ is non-zero, then the transition is *enabled* for $u_b(k), \delta_e(k)$. Besides, if more than one transition are enabled for $u_b(k), \delta_e(k)$, they are called *conflicting* on $(x_b(k), u_b(k), \delta_e(k))$. The finite state machine may impact the vehicle's states and operating mode (the example in Fig. 2 clarifies the connection between DHSA components).

In DHSA, the stochastic finite state machine can be replaced by a set of auxiliary binary variables $w_i(k)$ called *uncontrollable events*. Therefore, a DHSA can be converted to a discrete hybrid automata with uncontrollable events (ueHDA). The uncontrollable events for $l-1$ possible transitions are defined as (for $i = 1, \dots, l-1$)

$$P[w_i = 1] = p_i = f_{FSM}(x_b(k), u_b(k), \delta_e(k), \hat{x}_b),$$

which means that the i th transition $(x_b, u_b, \delta_e) \rightarrow \hat{x}_b$ happens if and only if $w_i = 1$. Furthermore, $w_l = 1$ implies that the transition is deterministic. If W_s represents the indices of the conflicting transitions on $x_b(k), u_b(k), \delta_e(k)$, the following equality should hold true for W_s

$$\sum_{i \in W_s} w_i(k) = 1.$$

If $\pi(k)$ denotes the probability of the transition occurred by $w(k)$, then the probability of a trajectory can be calculated using

$$\pi(\mathbf{w}_i) = \prod_{k=0}^{N-1} \pi(k), \quad (10)$$

where $\mathbf{w}_i$ is the vector of all uncontrollable events. By defining new auxiliary variables, a DHSA can be represented in the mixed logical dynamical (MLD) form (Bemporad & Morari, 1999). Rewriting (10) results in

$$\ln(\pi(\mathbf{w}_i)) = \sum_{k=0}^{N-1} \sum_{i=1}^l w_i(k) \ln(p_i). \quad (11)$$

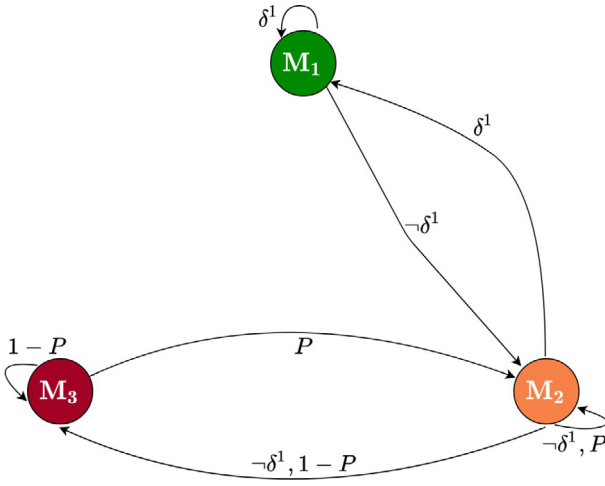


Fig. 2. Transition diagram for an illustrative example with 3 operating modes (M_1, M_2, M_3). $\neg \square$ represents negate of $\square$. δ^1 is the only event generator of the system. This system has two uncontrollable events: the first one occurs with the probability of P while the second one occurs with the probability of $1 - P$. The transition between modes M_2, M_3 depends on the uncontrollable events.

To eliminate trajectories with small probability, the following chance constraint is added

$$\ln(\pi(\mathbf{w}_i)) \geq \ln(\bar{p}),$$

where $0 \leq \bar{p} \leq 1$ is the probability bound. Hence, the cost function for a DHSA can be defined as

$$J(\mathbf{u}, \mathbf{w}, \mathbf{r}, \mathbf{x}(0)) = J_p - q \ln(\pi(\mathbf{w})), \quad (12)$$

where $\mathbf{u}$ is the vector of system inputs (within the prediction horizon), $\mathbf{r}$ is the vector of desired outputs, J_p is the performance index that can be chosen to be the l_2 or l_∞ norm, $-\ln(\pi(\mathbf{w}))$ is the probability cost, and the constant $q \geq 0$ is the probability cost weight. To better describe a DHSA system, an illustrative example is shown in Fig. 2. Beginning from mode M_1 , as far as $\delta^1 = 1$ ($\equiv \delta^1$ in the diagram), the system mode does not change. If $\delta^1 = 0$ ($\equiv \neg \delta^1$ in the diagram), then the system changes mode and moves to M_2 . In this example, two uncontrollable events exist: w_1 which occurs with the probability of P , and w_2 which occurs with the probability of $1 - P$. If $\delta^1 = 0$ while the current system mode is M_2 , then the uncontrollable events w_1 and w_2 specify the next mode of the system (if $w_1 = 1$, then system remains in mode M_2 ; otherwise the system will move to mode M_3).

The goal here is to minimize (12) subject to the system dynamics and constraints (including the chance constraints). Finally, the constrained-optimization problem, which is corresponding to the model predictive control design, will be solved using mixed-integer programming (see Bemporad & Di Cairano, 2010 for details).

3. Description of the vehicle model and operating modes in the mixed logical dynamical form

In the CACC problem described earlier, there are three operating modes: free-following, warning, and emergency braking. The uncontrollable events here are defined in terms of distance measurement error. DHSA is then used to model the system and the system dynamics are represented in the MLD form. Finally, mixed-integer quadratic programming can be employed to find the optimal control input(s). In the remainder of this section, system inequalities in the form of an MLD are derived.

The constraints on the system include bounds on the acceleration, input, road speed limit, and distance between vehicles (note that a negative distance implies collision and therefore should not occur). The

following inequalities should always hold true (for the simplicity of the notation, d_i and Δd_i are used in lieu of $d_{i,i-1}$ and $\Delta d_{i,i-1}$, respectively)

$$a_i^{min} \leq a_i(k) \leq a_i^{max}, \quad (13a)$$

$$u_i^{min} \leq u_i(k) \leq u_i^{max}, \quad (13b)$$

$$v_i(k) \leq v_{max}, \quad (13c)$$

$$d_i(k) > 0. \quad (13d)$$

As far as $\Delta v_i(k)$ is greater than a free-following threshold $\underline{v}^f < 0$, the vehicle operates the in free-following mode. However, if $\Delta v_i(k)$ goes below the free-following threshold, the i th vehicle enters either warning mode or emergency braking mode, which will be decided by the controller. Choosing a threshold based on the velocity allows vehicles to safely follow each other, especially if a deceleration occurs in the platoon. In this case, a CAV that moves faster than its preceding vehicle may enter warning/emergency-braking mode to prevent getting too close to its predecessor, thereby preserving safety. The auxiliary binary variable $g_i(k)$ is defined such that

$$\begin{aligned} \Delta v_i(k) \leq \underline{v}^f &\iff g_i(k) = 1, \\ g_i(k) = 1 &\iff \delta_i^w = 1 \text{ or } \delta_i^e = 1, \end{aligned} \quad (14)$$

where δ_i^e is the emergency braking event, and δ_i^w is the warning event. It is noted that $\delta_i^w = 1$ means that warning mode occurs while $\delta_i^e = 1$ means that the vehicle operates in emergency braking mode. To preserve safety, when a vehicle's warning/emergency-braking event is activated ($g_i = 1$), its target velocity will change from v_{i-1} to a smaller value $v_{i-1} - v_i^{we}$ while its target headway increases by $v_i^{we} t_s$, where v_i^{we} is a design parameter. The changes in the target velocity and headway will be taken into account in the cost function, as explained in Section 4.

As shown in Fig. 3, it is assumed that switching from free-following to warning/danger mode is probabilistic, and hence, when a CAV exits the free-following mode, as far as g_i is active ($g_i = 1$), the vehicle may enter warning mode with the probability P_w , or it may switch to emergency braking mode with the probability $P_e = 1 - P_w$. It is noted that P_w is also a control design parameter. In our simulation studies, P_w is considered to be 0.5, and hence the chance of entering warning or emergency-braking modes is equal from the controller's perspective. The representation of (14) in the MLD form is as follows:

$$\begin{aligned} \Delta v_i(k) - \underline{v}^f &\leq M_i^f [1 - g_i(k)], \\ \Delta v_i(k) - \underline{v}^f &\geq \epsilon + g_i(k) [m_i^f - \epsilon], \\ \delta_i^w(k) + \delta_i^e(k) &= g_i(k), \\ \delta_i^e(k+1) &\geq g_i(k+1) + \delta_i^e(k) - 1, \end{aligned} \quad (15)$$

where m_i^f and M_i^f denote lower and upper bounds on $\Delta v_i(k) - \underline{v}^f$, respectively. The last constraint in the above-mentioned equation indicates that if the vehicle enters the emergency braking mode, it will remain in that mode until $\Delta v_i > \underline{v}^f$, and the vehicle enters the free-following mode again.

3.1. Hard braking implementation

To enforce a hard braking in emergency braking mode, an upper bound constraint on $u_i(k)$ is added such that

$$u_i(k) \leq \delta_i^e(k) u_i^{min} + [1 - \delta_i^e(k)] u_i^{max}. \quad (16)$$

Thus, as far as the system is operating in the free-following mode, the upper bound on input is u_i^{max} . However, when emergency-braking mode is activated, the upper bound becomes u_i^{min} . Since the emergency-braking mode forces the input $u_i(k) = u_i^{min}$, the vehicle speed may become negative. To handle this issue, the constraint $v_i(k) \geq 0$ is added to the system. However, this may result in infeasibility because of two contradictory constraints; $v_i(k) \geq 0$ while $v_i(k+1) = v_i(k) + t_s a_i(k)$

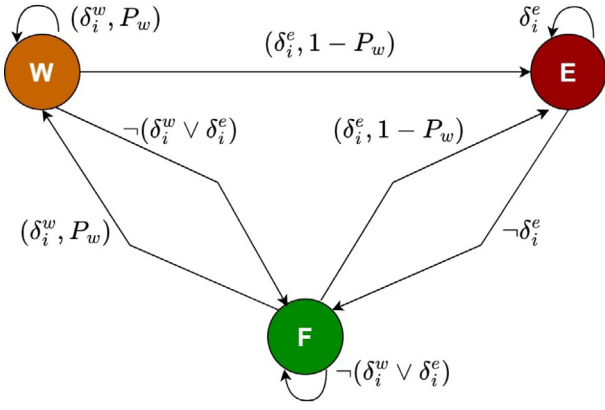


Fig. 3. CAV system operating modes diagram, where each CAV can operate in three modes: free-following mode (F), warning mode (W), and emergency-braking mode (E). Binary auxiliary variables δ_i^e and δ_i^w , and probability P_w specify the CAV's operating mode.

may become negative. To avoid this problem, a new binary variable is defined such that

$$v(k) < \underline{v}_i \iff \delta_i^v(k) = 0, \quad (17)$$

and (16) is rewritten as

$$u_i(k) \leq 0.5 [\delta_i^e(k) + \delta_i^v(k)] u_i^{\min} + [2 - \delta_i^e(k) - \delta_i^v(k)] \bar{u}_i, \quad (18)$$

where

$$\bar{u}_i \geq u_i^{\max} - 0.5 u_i^{\min}.$$

Inequality (18) implies that when emergency braking event is activated and $v_i(k) > \underline{v}_i$, the control input should be set to its lowest value. Besides, if $\delta_i^e(k) + \delta_i^v(k) \leq 1$, there is no need to enforce the vehicle input to get to its minimum, and the input is only bounded based on (13b). Hence, by choosing a small value for $\underline{v}_i$ (e.g., $\underline{v}_i = 1$ m/s), the issue of conflicting constraints is addressed. Eq. (17) in the MLD form turns into

$$v_i(k) - \underline{v}_i \leq M_i^v \delta_i^v(k),$$

$$v_i(k) - \underline{v}_i \geq \varepsilon + [1 - \delta_i^v(k)](m_i^v - \varepsilon),$$

where m_i^v and M_i^v are lower and upper bounds on $v_i(k) - \underline{v}_i$, respectively.

3.2. Measurement error consideration

Assuming that there is measurement error n_i in the model, it can be discretized (quantized) to different levels $\{c_i^1, c_i^2, \dots, c_i^{m_i}\}$ with known probabilities $\{p_i^1, p_i^2, \dots, p_i^{m_i}\}$. Therefore, it can be described as

$$n_i(k) = [c_i^1 \quad c_i^2 \quad \dots \quad c_i^{m_i}] \begin{bmatrix} w_i^1(k) \\ w_i^2(k) \\ \vdots \\ w_i^{m_i}(k) \end{bmatrix},$$

where $w_i^j(k)$, $j \in \{1, 2, \dots, m_i\}$ are auxiliary binary variables that represent uncontrollable events and $P[w_i^j(k) = 1] = p_i^j$. Based on the above parametrization of the noise, we have

$$\sum_{j=1}^{m_i} w_i^j(k) = 1.$$

Using the auxiliary variables added to the system, the discrete-time state space model in (7) is rewritten as

$$x_i(k+1) = (I + t_s A_i) x_i(k) + t_s B_i u_i(k) + t_s D_i a_i^m(t) + C_i W_i(k), \quad (19)$$

where

$$W_i(k) = [w_i^1 \quad w_i^2(k) \quad \dots \quad w_i^{m_i}(k)]^T, \\ C_i = \begin{bmatrix} c_i^1 & c_i^2 & \dots & c_i^{m_i} \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 \end{bmatrix}.$$

The transition diagram for each CAV with three operating modes is illustrated in Fig. 3. It is noted that the transition between the modes depends on the auxiliary binary variables (δ_i^e , δ_i^w) and the probability P_w defined earlier.

4. Formulation of the discrete hybrid stochastic MPC problem for CACC design

In the CAV platoon, our goal is that the distance between CAVs converges to its desired value while vehicles move with equal and constant speed. MPC has shown to have the capability of controlling multi-input multi-output systems. However, for the CACC problem, when the number of vehicles increases, centralized MPC is not time-efficient (Zhou et al., 2019). Instead, distributed MPC can be used to reach the desired performance. The MPC problem for CAV_i , which employs an m -vehicle-ahead communication topology, at time t is defined as follows

$$\min_{\mathbf{u}_i, \mathbf{w}_i, \mathbf{z}_i} \sum_{k=0}^{N-1} \left((x_i(k) - R_i)^T Q_i (x_i(k) - R_i) \right) - q_i^p \ln(\pi(\mathbf{w}_i)) \\ \text{subject to: } \text{MLD system equations,} \\ \ln(\pi(\mathbf{w}_i)) \geq \ln(\bar{p}_i), \quad (20)$$

where $\mathbf{u}_i$ and $\mathbf{z}_i$ are the system inputs and the vector of auxiliary variables from $k = 0$ to $k = N - 1$, respectively, and the constant q_i^p is the probability cost weight.

Each vehicle sends its current location, current speed, and current and predicted acceleration information every t_c seconds. To avoid unnecessary data exchange, it is assumed that $t_c \geq t_s$. Hence, each vehicle uses the last received data to solve (20) until the preceding vehicle shares new information. That means each vehicle solves its MPC problem every t_s seconds and finds $\mathbf{u}_i^*$, $\mathbf{x}_i^*$, and $\mathbf{w}_i^*$. Then, it discards $\mathbf{w}_i^*$ and applies the first sample of $\mathbf{u}_i^*$ to the system. Assuming that the vehicles use four predecessor-following communication scheme ($m = 4$), Q_i 's used in the simulation studies are considered as

$$Q_1 = \text{diag}[3 \ 3 \ 0.35],$$

$$Q_2 = \text{diag}[3 \ 0.25 \ 3 \ 1 \ 0.35],$$

$$Q_3 = \text{diag}[3 \ 0.25 \ 0.18 \ 3 \ 1 \ 0.70 \ 0.35],$$

$$Q_i = \text{diag}[3 \ 0.25 \ 0.18 \ 0.14 \ 3 \ 1 \ 0.70 \ 0.55 \ 0.35], \quad \forall i \geq 4.$$

For our CACC problem, the vector R_i is defined as $R_i = [g_i(k) \ v_i^{ue} \ t_s \ \mathbf{1}_{1 \times m} \ g_i(k) \ v_i^{ue} \ \mathbf{1}_{1 \times m} \ 0]^T$. This definition enforces CAV_i to drive with a smaller velocity and larger target headway compared to the m predecessor vehicles when CAV_i does not operate in the free-following mode. For relatively large time gaps (e.g., $h = 0.7$ s), v_i^{ue} is chosen around $0.01 \ v_i(t)$ while for smaller gaps (e.g., $h = 0.4$ s) v_i^{ue} is chosen around $0.1 \ v_i(t)$, where $v_i(t)$ is CAV_i velocity at time t .

Remark 2. According to Zhou et al. (2019), a platoon of N vehicles is l_∞ string-stable if and only if

$$\|\Delta d_i\|_{l_\infty} \leq \|\Delta d_{i-1}\|_{l_\infty}, \quad \forall i \in \{1, 2, N-1\}. \quad (21)$$

The equation above implies that in an l_∞ string-stable platoon, the peak magnitude deviation from equilibrium is not amplified through the platoon. It is noted that in order to guarantee the system's l_∞ string stability, cost and state constraints should be carefully designed. System constraints are also relaxed in the design process (Zhou et al.,

2019). Based on our formulated hybrid model in this paper, if system constraints are relaxed, the hybrid model would turn into a standard (i.e., non-hybrid) model. In this case, the discussions and proofs of string stability proposed in Zhou et al. (2019) for MPC controller will hold for our system as well. Besides, if a string-stable solution becomes infeasible, string stability and terminal constraints will be relaxed due to the importance of safety constraints and physical limits (Zhou et al., 2019). In this case, the constraints introduced in the previous section, including those that turn our model into a hybrid model, are activated to assure safety (e.g., performing emergency braking). It is noted that simulation results in the next section also demonstrate the string stability of the platoon with the proposed DHSA-MPC approach.

Remark 3. In the proposed MPC controller design approach, CAVs leverage the multi-predecessor-following communication structure to alleviate the impact of potential communication losses on their performance. With multi-predecessor-following communication, even if a CAV loses data communication with its preceding vehicle, it is still able to use other predecessors' predictive information to adjust its speed. The imperfect communication may still, however, put the first few follower vehicles (mainly the two vehicles right behind the leader) at risk since they only receive predictive information from a limited number of vehicles. But, as shown in the next section, integrating three operating modes in the controller design assures safety when predictive information is not available, especially when the leader vehicle drastically decelerates — they may end up stopping at a slightly smaller standstill distance from their immediate predecessors compared to the rest of the vehicles in the platoon.

5. Simulation results and discussion

For the CACC problem, performance of the proposed DHSA-MPC is compared against two control approaches, namely a proportional derivative (PD) controller, and a regular (non-hybrid) MPC scheme. Parameters used in our simulations are given in Table 1. CVXPY package in Python is used for finding the control input for regular MPC and DHSA-MPC, and Gurobi optimization package is used as the solver for the mixed-integer programs (Agrawal, Verschuere, Diamond, & Boyd, 2018; Diamond & Boyd, 2016; Gurobi Optimization, LLC, 2021) while PD controller is implemented using Simulink (Simulink Documentation, 2021). It is noted that the average computational time to solve the mixed-integer program (20) using an 11th Gen. Intel(R) Core(TM) i7-11800H @ 2.30 GHz laptop is under 5 ms. The computational time for the follower vehicles in a platoon, including five vehicles that use DHSA-MPC, is shown in Fig. 4. A normal distribution with zero mean and variance of 0.08 is considered for the measurement error, which is discretized to 11 levels, i.e., $n_i \in \{-0.25, -0.20, \dots, 0.25\}$ (as shown in Fig. 5) for the DHSA-MPC. First, the impact of the communication loss on performance of the PD controller and DHSA-MPC design is investigated. Then, the DHSA-MPC behavior considering low-rate intermittent communication with delay is studied. Next, the impact of the probability weight (q_i^p) on the CAVs' performance is examined. Finally, the performance of the vehicles controlled by DHSA-MPC is compared against the regular (non-hybrid) MPC considering low-rate intermittent communication with delay. In Figs. 6–8 and 10, the first subplot shows $d_i(t)$, the second subplot depicts vehicles' velocities $v_i(t)$, and the last subplot shows each vehicles' acceleration profiles.

5.1. Comparison between PD controller with DHSA-MPC controller during communication loss

In the first set of simulation studies, it is assumed that $h = 0.7$ s, $v^f = 2$ m/s, $t_c = 0.1$ s, and the communication between the leader and its follower is lost at $30 \text{ s} \leq t \leq 30.4 \text{ s}$. It is also assumed that vehicles use a one-predecessor-following communication topology for

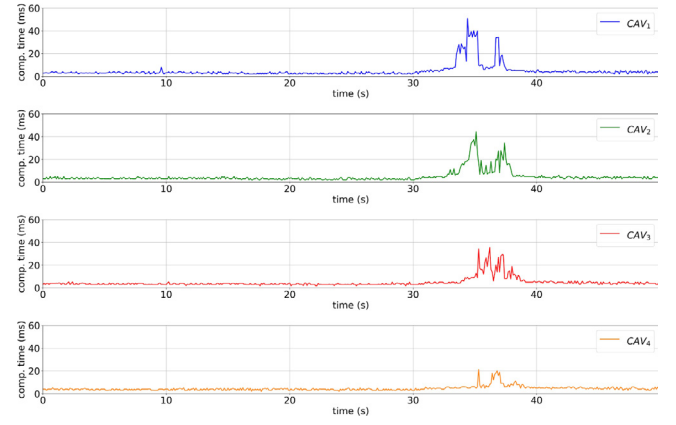


Fig. 4. Computational time for the follower vehicles in a five-vehicle platoon using DHSA-MPC during a 50 s simulation. The average computational time for a vehicle is under 5 ms. The computational time for vehicles increases during a sudden deceleration event, which occurs after $t = 30$ s, since the vehicles' operating mode may change from free following to warning or emergency braking.

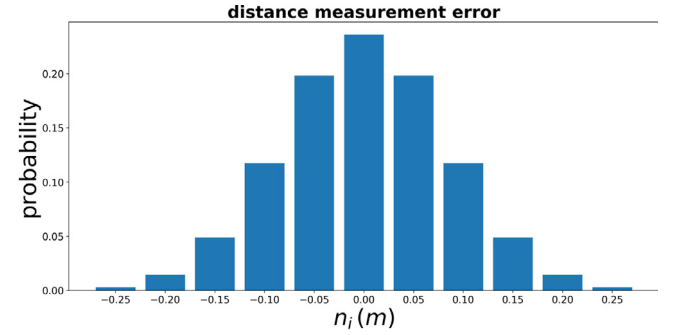


Fig. 5. Discretized measurement error probability distribution.

Table 1

Model and optimization parameters used in the simulations.

Parameter	Value	Parameter	Value
t_s	0.1 s	m	4
h	0.7 s, 0.4 s	τ_i	0.1 s
l_i	5 m	r_i	2 m
v^f	2 m/s, 0.5 m/s	N	7
a_i^{min}	-4 m/s^2	a_i^{max}	3 m/s^2
u_i^{min}	-4 m/s^2	u_i^{max}	4 m/s^2
$\hat{p}$	0.01^N	q_i^p	0.60

the PD controller while the DHSA-MPC employs a four-predecessor-following communication structure. It is noted that previous assumption does not affect the comparison since the first follower in the platoon only receives information from the platoon's leader and the aforementioned communication loss occurs between the leader and its immediate follower. Although the PD controller performs well in tracking the preceding vehicle, it is highly dependent on the proper V2V communication. As shown in Fig. 6, the PD controller is not able to preserve safety, and accident occurs when the communication is lost for a relatively short time (in the first subplot in Fig. 6, the distance between the first vehicle and the leader becomes negative at around $t = 37$ s). As observed, the fluctuations in the acceleration plot are because of the noisy vehicle distance measurements ($d_i(t)$). The DHSA-MPC, which leverages the predictive information received through communication from multiple predecessors is, however, able to preserve safety of the vehicle platoon while showing a noticeably strong tracking performance with a smooth acceleration profile for vehicles (see Fig. 7). It is noted that integrating emergency-braking and warning modes into controller design prevents performing unnecessary braking while allows vehicles to properly keep

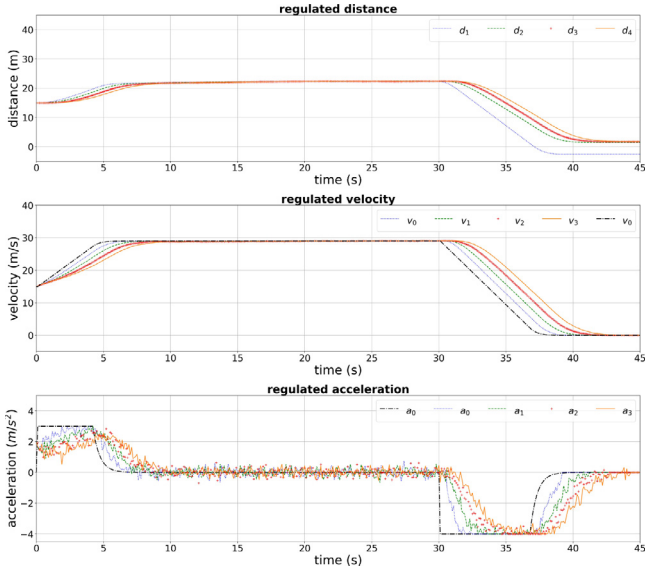


Fig. 6. PD controller results for a platoon of 5 vehicles. The noisy behavior in the acceleration plot is due to the distance measurement error. Between times 30 s and 30.4 s, the communication between the leader and the first vehicle is lost which results in an accident in the platoon at around $t = 37$ s.

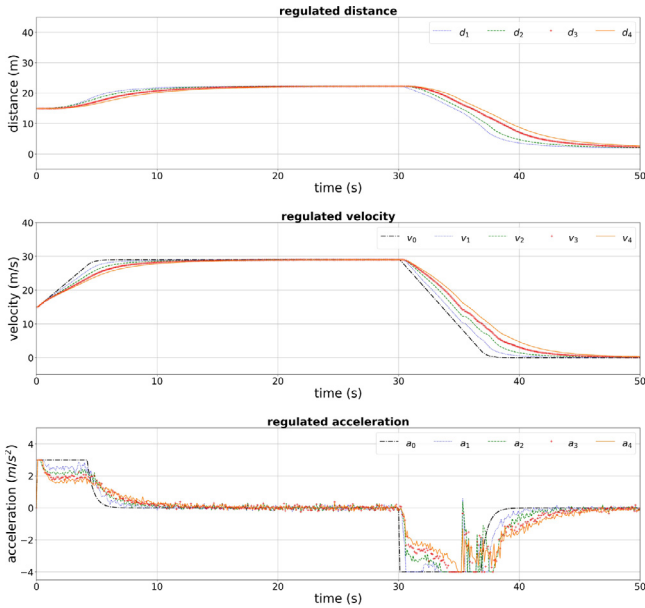


Fig. 7. DHSA-MPC control results for a platoon of 5 vehicles with $t_c = t_s$. Although the communication between the first two vehicles is lost from $t = 30$ s to $t = 30.4$ s, the controller is successfully able to preserve safety.

their headway d_i close to the desired value d_i^* even during sudden speed changes. Comparing the acceleration profiles of the PD controller and the DHSA-MPC reveals that the latter is able to reduce the impact of noisy distance measurement on the vehicle acceleration profile. According to the results, vehicles converge to their desired headway for PD controller at around 7 s while for DHSA-MPC, this time is around 11 s. In DHSA-MPC, vehicles converge to their desired headway slower since the convergence rate is sacrificed to maintain safety.

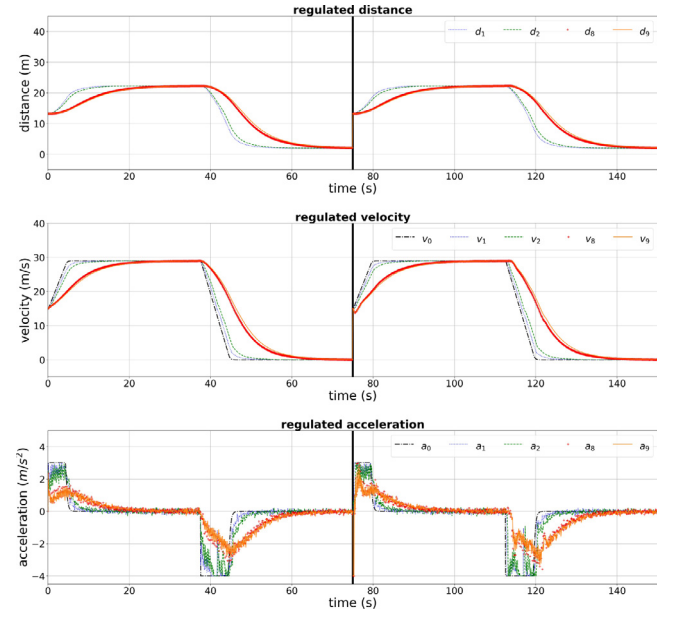


Fig. 8. DHSA-MPC control results for a platoon of 10 vehicles with a four-predecessor-following communication topology for two cases (with no communication delay from time 0 s to 75 s, and from 75 s to 150 s with communication delay). In both cases, communication rate is $t_c = 0.3$ s while Bernoulli loss with the probability of 0.1 is used. The performance of the four follower vehicles, along with the leader vehicle, is depicted for better comparison. Comparing the results reveals that even with the low-rate intermittent communication and delay, the controller is able to preserve safety, and each CAV successfully tracks its preceding vehicle. The considerable fluctuation in acceleration is due to the intermittent and low-rate communication.

5.2. Impact of communication on the DHSA-MPC's performance

In the second study, DHSA-MPC (that considers a four-predecessor-following communication scheme) is employed for two different communication cases; the first case in which no communication delay is considered, and the second case in which the delay of either 0.1 s, 0.2 s, 0.3 s is considered. In both cases, $v^f = 2$ m/s, and $t_c = 0.3$ s are used, and a Bernoulli communication loss with the probability of 0.1 is considered. As shown in Fig. 8, the simulation results from $t = 0$ s to $t = 75$ s show the vehicles' behavior when communication delay is zero, while after resetting the simulation environment, the vehicles' performance in the presence of communication delay is depicted from $t = 75$ s to $t = 150$ s. Simulation results show that the DHSA-MPC performance is satisfactory even in the presence of communication delay and loss. Noticeable changes in the follower vehicles' acceleration are observed as a result of choosing $t_c > t_s$ and intermittent communication. When $t_c > t_s$, by approaching the next communication instant, each vehicle has less information available about the predecessor vehicles' future decisions, and before the next communication instant, each vehicle mostly has the past information about the preceding vehicle's acceleration. Employing multi-predecessor-following communication allows vehicles to detect changes in the predecessors' behavior relatively quickly and help vehicles act accordingly. It is noted that the standstill distance for the first two follower vehicles is slightly less than its desired value according to Remark 2.

5.3. The probability weight constant value's impact on the CAV's performance

The third simulation study focuses on the impact of the probability weight constant q_i^p on the CAVs' performance for a 5-vehicle platoon. As shown in Fig. 9, first to fourth subplots depict follower CAVs' acceleration and the last subplot shows Δd_i for vehicles after all vehicles' velocity converge to their desired value (which is the leader's

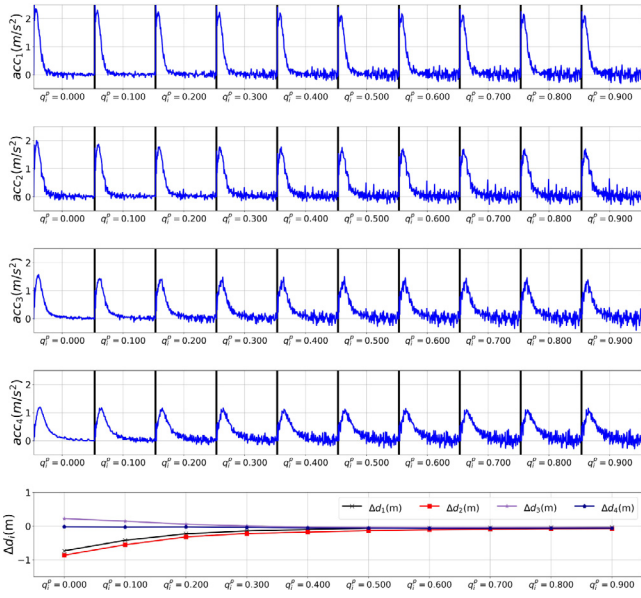


Fig. 9. Figure shows the impact of the probability weight constant q_i^p on the vehicles' performance in a five-vehicle platoon. As observed, increasing q_i^p adds a noisy behavior to the vehicle acceleration while reducing the error between the vehicle headway and its desired value.

speed). According to the results, increasing q_i^p from zero increases CAVs' acceleration fluctuations while reducing Δd_i . Hence, a small value of q_i^p reduces the impact of the distance measurement noise on the CAV acceleration profile while it may result in an error in tracking the desired spacing policy. This error is negligible for the fourth CAV since the multi-predecessor-following communication scheme diminishes the impact of the distance measurement noise on Δd_i .

5.4. Comparison between DHSA-MPC and regular (non-hybrid) MPC considering communication loss and delay

In the last simulation scenario, the performance of the MPC in the presence of a Bernoulli communication loss with the probability of 0.1 and communication rate of 0.3 s is studied. Also, communication delay of 0.2 s and time gap of $h = 0.4$ s is considered for both cases. In the first case, the hybrid design (DHSA-MPC) is taken into account (Fig. 10, $t = 0$ s to $t = 50$ s) while in the second case, a non-hybrid MPC is used (Fig. 10, $t = 50$ s to $t = 84$ s). In both cases, $h = 0.4$ s and $v^f = 0.5$ m/s. As shown in Fig. 10, DHSA-MPC is successfully able to handle intermittent communication and sudden deceleration, thanks to the warning and emergency braking modes built in the control design. Vehicles, in this case, reach their desired standstill distance, which is $r_i = 2$ m. However, regular MPC cannot preserve safety during deceleration, and accidents occur in the platoon at around $t = 83.4$ s. Results confirm the benefit of considering multiple operating modes to preserve safety at all times.

6. Concluding remarks

In this paper, a new stochastic MPC design method has been developed based on a discrete hybrid stochastic model for CACC applications aiming at safe platooning. It is assumed that each CAV can measure the distance from its immediate predecessor, as well as the predecessor's velocity. Through communication, it also has access to a limited number of predecessors' location, speed, and predictive acceleration profile. Since vehicles do not need other vehicles' model, the proposed MPC design method can be applied to a heterogeneous fleet of vehicles in a platoon. The proposed MPC scheme can take imperfect sensing information, adapt to low-rate intermittent vehicle-to-vehicle communication,

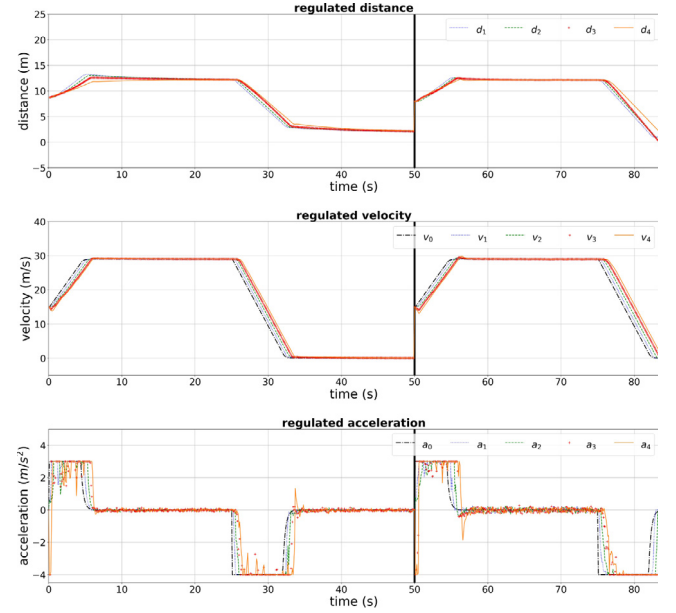


Fig. 10. Comparison between the performance of the DHSA-MPC ($t = 0$ s to $t = 50$ s) and the regular (non-hybrid) MPC ($t = 50$ s to $t = 84$ s) in the presence of communication loss and delay. While DHSA-MPC is able to successfully and safely satisfy the desired performance, the regular MPC is not able to preserve safety during sudden deceleration, and accident occurs at around $t = 83.4$ s.

and integrate three operating modes for each vehicle, namely, free following, warning, and emergency braking. Each vehicle operating mode is chosen based on the predictive information it receives from its predecessors, based on which it adjusts its speed trajectory to avoid unnecessary hard braking. The proposed stochastic controller is successfully able to counteract the effect of imperfect sensing information by providing a smooth acceleration profile. It is shown that leveraging the multi-predecessor-following communication topology improves the vehicles' performance as CAVs can detect changes in the platoon as quickly as possible. It is noted that while slow and imperfect data communication may cause fluctuations in the CAV acceleration/deceleration, the proposed method is shown to still preserve safety in the platoon. The proposed approach works successfully in the presence of communication delay and loss as well.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was financially supported by the US National Science Foundation under award number #1931981.

References

- Abou Harfouch, Y., Yuan, S., & Baldi, S. (2017). An adaptive switched control approach to heterogeneous platooning with intervehicle communication losses. *IEEE Transactions on Control of Network Systems*, 5(3), 1434–1444.
- Agrawal, A., Verschueren, R., Diamond, S., & Boyd, S. (2018). A rewriting system for convex optimization problems. *Journal of Control and Decision*, 5(1), 42–60.
- Bemporad, A., & Di Cairano, S. (2010). Model-predictive control of discrete hybrid stochastic automata. *IEEE Transactions on Automatic Control*, 56(6), 1307–1321.
- Bemporad, A., & Morari, M. (1999). Control of systems integrating logic, dynamics, and constraints. *Automatica*, 35(3), 407–427.
- Cui, L., Chen, Z., Wang, A., Hu, J., & Park, B. B. (2021). Development of a robust cooperative adaptive cruise control with dynamic topology. *IEEE Transactions on Intelligent Transportation Systems*.

- Diamond, S., & Boyd, S. (2016). CVXPY: A Python-embedded modeling language for convex optimization. *Journal of Machine Learning Research*, 17(83), 1–5.
- Guo, G., & Yue, W. (2014). Sampled-data cooperative adaptive cruise control of vehicles with sensor failures. *IEEE Transactions on Intelligent Transportation Systems*, 15(6), 2404–2418.
- Gurobi Optimization, LLC (2021). Gurobi optimizer reference manual. <http://www.gurobi.com>.
- Hasan, S., Balador, A., Girs, S., & Uhlemann, E. (2019). Towards emergency braking as a fail-safe state in platooning: A simulative approach. In *2019 IEEE 90th vehicular technology conference* (pp. 1–5). IEEE.
- Kianfar, R., Falcone, P., & Fredriksson, J. (2015). A control matching model predictive control approach to string stable vehicle platooning. *Control Engineering Practice*, 45, 163–173.
- Lan, J., Zhao, D., & Tian, D. (2020). Robust cooperative adaptive cruise control of vehicles on banked and curved roads with sensor bias. In *2020 American control conference* (pp. 2276–2281). IEEE.
- Lee, K. K., & Chanson, S. T. (2002). Packet loss probability for real-time wireless communications. *IEEE Transactions on Vehicular Technology*, 51(6), 1569–1575.
- Morbidi, F., Colaneri, P., & Stanger, T. (2013). Decentralized optimal control of a car platoon with guaranteed string stability. In *2013 European control conference*, (pp. 3494–3499). IEEE.
- Moser, D., Waschl, H., Kirchsteiger, H., Schmied, R., & Del Re, L. (2015). Cooperative adaptive cruise control applying stochastic linear model predictive control strategies. In *2015 European control conference* (pp. 3383–3388). IEEE.
- Murthy, D. K., & Masrur, A. (2016). Braking in close following platoons: The law of the weakest. In *2016 Euromicro conference on digital system design* (pp. 613–620). IEEE.
- Ploeg, J. (2014). Analysis and design of controllers for cooperative and automated driving.
- Ploeg, J., Semsar-Kazerooni, E., Lijster, G., van de Wouw, N., & Nijmeijer, H. (2014). Graceful degradation of cooperative adaptive cruise control. *IEEE Transactions on Intelligent Transportation Systems*, 16(1), 488–497.
- Simulink Documentation (2021). Simulation and model-based design. URL <https://www.mathworks.com/products/simulink.html>.
- Trudgen, M., Miller, R., & Velni, J. M. (2018). Robust cooperative adaptive cruise control design and implementation for connected vehicles. *International Journal of Automation and Control*, 12(4), 469–494.
- Turri, V., Besselink, B., & Johansson, K. H. (2016). Cooperative look-ahead control for fuel-efficient and safe heavy-duty vehicle platooning. *IEEE Transactions on Control Systems Technology*, 25(1), 12–28.
- Vahidi, A., & Eskandarian, A. (2003). Research advances in intelligent collision avoidance and adaptive cruise control. *IEEE Transactions on Intelligent Transportation Systems*, 4(3), 143–153.
- Van Arem, B., Van Driel, C. J., & Visser, R. (2006). The impact of cooperative adaptive cruise control on traffic-flow characteristics. *IEEE Transactions on Intelligent Transportation Systems*, 7(4), 429–436.
- Van Nunen, E., Verhaegh, J., Silvas, E., Semsar-Kazerooni, E., & Van De Wouw, N. (2017). Robust model predictive cooperative adaptive cruise control subject to V2V impairments. In *2017 IEEE 20th international conference on intelligent transportation systems* (pp. 1–8). IEEE.
- Wang, Z., Wu, G., & Barth, M. J. (2018). A review on cooperative adaptive cruise control (CACC) systems: Architectures, controls, and applications. In *2018 21st international conference on intelligent transportation systems* (pp. 2884–2891). IEEE.
- Wu, C., Lin, Y., & Eskandarian, A. (2019). Cooperative adaptive cruise control with adaptive Kalman filter subject to temporary communication loss. *IEEE Access*, 7, 93558–93568.
- Zhai, C., Liu, Y., & Luo, F. (2018). A switched control strategy of heterogeneous vehicle platoon for multiple objectives with state constraints. *IEEE Transactions on Intelligent Transportation Systems*, 20(5), 1883–1896.
- Zhou, Y., Wang, M., & Ahn, S. (2019). Distributed model predictive control approach for cooperative car-following with guaranteed local and string stability. *Transportation Research, Part B (Methodological)*, 128, 69–86.