



## **Buffer-Aided Model Predictive Controller to Mitigate Model Mismatches and Localization Errors**

Downloaded from: <https://research.chalmers.se>, 2025-04-26 18:47 UTC

Citation for the original published paper (version of record):

Patel, R., Wymeersch, H., Härrä, J. et al (2018). Buffer-Aided Model Predictive Controller to Mitigate Model Mismatches and Localization Errors. *IEEE Transactions on Intelligent Vehicles*, 3(4): 501-510. <http://dx.doi.org/10.1109/TIV.2018.2873908>

N.B. When citing this work, cite the original published paper.

© 2018 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, or reuse of any copyrighted component of this work in other works.

# Buffer-aided Model Predictive Controller to Mitigate Model Mismatches and Localization Errors

Raj Haresh Patel, *Student Member, IEEE*, Henk Wymeersch, *Member, IEEE*, Jérôme Härrri, *Member, IEEE*,  
Christian Bonnet, *Member, IEEE*

**Abstract**—Any vehicle needs to be aware of its localization, destination and neighboring vehicles’ state information for collision free navigation. A centralized controller computes controls for Cooperative Adaptive Cruise Control (CACC) vehicles based on the assumed behavior of manually driven vehicles (MDVs) in a mixed vehicle scenario. The assumed behavior of the MDVs may be different from the actual behavior which gives rise to a model mismatch. The use of erroneous localization information can generate erroneous controls. The presence of a model mismatch and the use of erroneous controls could potentially result into collisions. A controller robust to issues like localization errors and model mismatches is thus required. This paper proposes a robust model predictive controller which accounts for localization errors and mitigates model mismatches. Future control values computed by the centralized controller are shared with CACC vehicles and are stored in a buffer. Due to large localization errors or model mismatches when control computations are infeasible, control values from the buffer are used. Simulation results show that the proposed robust controller with buffer can avoid almost the same number of collisions in a scenario impacted by localization errors as that in a scenario with no localization errors despite model mismatch.

**Index Terms**—Robust Model Predictive Control, Localization Errors, Model Mismatch, Centralized Control.

## I. INTRODUCTION

Improvements in sensing, control and localization techniques along with higher computational capabilities have facilitated the entry of automation in vehicular domain. Cooperative Adaptive Cruise Control (CACC) is an operational feature of a vehicle with cruise control and vehicle to everything (V2X) communication capability [1], which leverages onboard V2X communication capability to communicate and coordinate with other CACC vehicles. Controls of such CACC enabled vehicles (referred to as CACC vehicles hereon) can either be computed locally in a decentralized control system or remotely on either a road side unit (RSU) or a centralized controller if they are a part of a centralized control system. While driving, vehicles need to accelerate, brake, maintain constant velocity, etc. More specifically, behavior of vehicles

performing such actions can be studied under different scenarios like ramp merging, while approaching an intersection, during freeflow on a highway, etc. A critical scenario is the braking scenario, where multiple vehicles on the same lane are notified of an emergency event ahead and vehicles must brake and come to a halt within a particular distance or time.

Vehicles use different localization techniques with varying performances. Localization based on GPS usually has a standard deviation (std) of error around 4 m and the accuracy depends on various factors like the environment, satellites in line of sight, etc. [2]. Map matching techniques implemented on Waymo’s autonomous vehicle achieve localization with std of accuracy better than 10 cm [3]. HIGHTS is an example of an European project with a goal to achieve high precision positioning system with the accuracy of 25 cm for Intelligent Transportation System (ITS) [4]. Localization errors translate to control input errors, which in turn can lead to collisions. In order to provide controls for collision free maneuvers, it is necessary to account for localization errors.

In a mixed vehicle scenario, consisting of CACC and manually driven vehicles (MDVs), human factors like limited visibility and perception response time influence human’s driving behavior [5] which can not be accurately predicted. Although online estimation and modeling of neighboring vehicles can be performed [6]–[8] it might be computationally intensive. Control behavior generated from such an estimated model can still be different from the actual human behavior. This difference between the estimated model (assumed behavior) of a human driver and the actual model of a human driver controlling a MDV is termed as a model mismatch in this paper. Controls computed using estimated state parameters different from the actual state parameters could lead to collisions. Thus it is necessary to mitigate model mismatch. The presence of constantly changing localization error and model mismatch necessitates recomputation of controls for CACC vehicles. This is the basis of a model predictive control (MPC) where controls are computed at each time slot using updated state parameters in a receding horizon manner.

The main contribution of this work is proposing a centralized controller robust to localization uncertainties and model mismatches. It consists of three key components: 1) a MPC controller: to provide robustness against model mismatch 2) an algorithm that accounts for localization errors: to mitigate the impact of localization error 3) a buffer: to store the future intended controls computed by the controller; in case of

Raj Haresh Patel, Jérôme Härrri and Christian Bonnet are with the Communication Systems Department, EURECOM, France. E-mail: {patel,haerri,bonnet}@eurecom.fr. Henk Wymeersch is a member of Communication Systems group, Department of Electrical Engineering, Chalmers University of Technology, Gothenburg, Sweden. E-mail: henkw@chalmers.se. This work is supported, in part, by a PhD Grant from the Graduate School of the University Pierre Marie Curie (UPMC), Paris. EURECOM acknowledges the support of its industrial members, namely BMW Group, IABG, Monaco Telecom, Orange, SAP, ST Microelectronics and Symantec.

infeasibility of control computation (due to model mismatch or localization error), controls from the buffer can be fetched. The proposed robust MPC controller is evaluated in a mixed vehicle braking scenario and the use of each component is validated.

The remainder of this paper is structured as follows. In Section II related work is discussed. Section III introduces the problem statement and concept of model mismatch. The MDV model assumed by the centralized controller, the algorithm that counters localization errors and the use of buffer to counter infeasibility issues of the proposed MPC based controller are described in Section IV. Simulations are analyzed in Section V and concluding thoughts are presented in Section VI.

## II. RELATED WORK

There can be multiple sources of model mismatch. According to Chen et al. communication and actuator delays results into different values of vehicles' state parameters compared to the actual values leading to a model mismatch [9]. If MDVs are modeled using IDM+ (Intelligent Driving Model + was introduced in [10]), model mismatches can also arise from uncertainties or unknown parameters of IDM+ [9]. Uncertainty in range of frequency responses from engine torque command to speed can also be modeled as a model mismatch [11]. Aramrattana et al. evaluated the performance of a centralized controller under model mismatch arising due to the difference in the assumed controls and the controls obtained from real humans used in their simulations [12].

The presence of model mismatch and localization error necessitates frequent computation and transmission of controls to CACC vehicles. The frequency of transmission of controls could be optimized based on the magnitude of localization error [13]. To evaluate robustness of centralized controller to localization errors authors add a random value of distance disturbance during simulations [14]. Patel et al. showed unaccounted localization error results into a reduction in flow capacity and leads to accidents in a centralized control system; they also proposed an algorithm to counter localization error [15], [16]. [13]–[16] have two main drawbacks. First, it does not account for space/time varying localization errors. Second, strong localization errors leading to computational infeasibility were simply profiled as 'failure/accident'.

MPC-based controllers have been widely used in the literature. A centralized MPC based optimization algorithm for intersection clearance is used to optimize the longitudinal motion of a vehicle by researchers at University of California Berkeley [17]. Rodrigues de Campos et al. evaluate centralized control algorithm vs decentralized control algorithm for traffic coordination at intersections [18]. More examples of MPC based centralized and decentralized controllers can be found in a literature survey by Rios-Torres et al. [19]. Wang et al. focus on longitudinal collision avoidance based on MPC for centralized coordinated braking [20]. Similar work has been accomplished by Lu and other researchers from PATH, University of California, Berkeley [21]. Both [20], [21] focus on longitudinal collision avoidance via coordinated braking by minimizing kinetic energy between pairs of autonomous vehi-

cles (or strictly MDVs) assuming ideal communication and localization capability. In order to make MPC-based controllers perform well even under adversaries, robust MPC techniques are implemented. A min-max MPC approach has been used to make a centralized controller robust to uncertainties in MDV behavior [17] where as Chen et al. use min-max MPC to create a centralized controller robust to communication and actuator delays [9]. Authors Gao et al. in [11] propose the use of a robust controller to counter both parameter uncertainties and uniform communication delay. The use of a buffer to mitigate the impact of communication failures has been recently proposed for a decentralized controller [22]. In short, papers [9], [11], [17], [22] propose robust algorithms to counter path planning uncertainties, communication uncertainties, actuator delays and vehicle model uncertainties like uncertain range of frequency responses from torque command respectively.

In this paper we focus on a mixed vehicle coordinated braking scenario in the presence of model mismatch and localization errors. Model mismatch is realized using different models for assumed and actual MDV controls and the lower level controller is assumed to be perfect. Localization uncertainty is modeled as space and time varying errors. The centralized controller is made robust first by integrating localization errors in the MPC controller and second by including a buffer in order to avoid the lack of controls in case of a computational infeasibility.

## III. SYSTEM MODEL

In this paper we consider longitudinal motion of multiple vehicles on a single lane containing *CACC enabled* vehicles and *MDV* as illustrated in Fig. 1. We assume a MDV is driven by a human and is without any control capabilities. On the other hand, CACC vehicles are assumed to start implementing control action simultaneously on the reception of controls inputs from the centralized controller. The frequency of received control inputs is defined by the controller's update frequency. Without loss of generality, we set the update frequency to 10 Hz. This is motivated by the fact that both the maximum realizable rate of steering commands [23] and the state-of-art GPS-fix update rate both are around 10 Hz.

### A. Informal Problem Statement

We consider a system of multiple vehicles traveling in the same direction on a single lane, as illustrated in Fig. 1. Each vehicle is characterized by its position and velocity. If the first vehicle has to brake on sensing an obstacle ahead, the following vehicles would need to brake as well in a coordinated way. There exists a centralized control system like a Cloud or Edge Service, located either in the Cloud or in ITS-G5 or Cellular infrastructures. All vehicles (CACC and MDVs) are connected and transmit their noisy position, speed and acceleration estimates to the centralized controller. The red ellipses in Fig. 1 depict the MDVs' and CACC vehicles' localization errors impacting the centralized controller. This information is used by the centralized controller to compute control inputs, which are subsequently sent back to CACC vehicles. MDVs react as function of the vehicle in front,

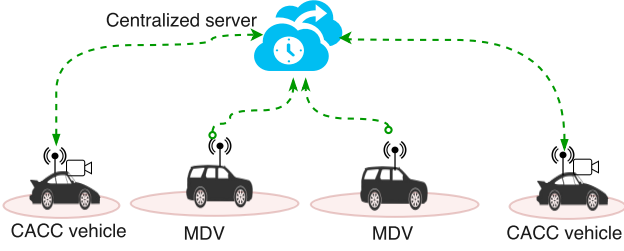


Fig. 1. Centralized control of CACC vehicles in a mixed vehicle scenario.

according to the model described in Section III-B. The central controller accounts for the MDVs through simple but realistic models.

In summary, the objective is for the central controller to bring all vehicles to a complete stop without collision with minimal discomfort, in the presence of localization errors and model mismatch.

### B. Actual Model of MDV

The actual behavior of MDV is based on ‘follow-the-leader’ strategy. In order to imitate actual human driving, MDVs are assumed to respond and react to the vehicle in front after a certain time delay represented by the perception response time. If there are multiple MDVs following one another in a stream of vehicles, the effective perception reaction time of a MDV would be proportional to the number of MDVs immediately ahead. We refer readers to [16] and references therein for a detailed explanation. This effective perception response time (now on referred to as perception response time) for vehicle  $i$  is denoted by  $t_{i,1}$ . The behavior of MDVs after perception response time is actually governed by intelligent driver model (IDM) [24], i.e., a simplified version of Human Driving Model [25] where we assume the visibility to be limited to just the vehicle in front. We consider time slots of duration 100 ms, and introduce  $c = 10$  slots/second. Control input is assumed to be the acceleration of the vehicle  $u_i(n)$ . The profile implemented by MDVs can be represented mathematically as follows: for time slots  $n \leq c \cdot t_{i,1}$  (i.e., before the perception response time),  $u_i(n) = 0$ . For  $n > c \cdot t_{i,1}$ ,

$$u_i(n) = u_i^{\max} \left( 1 - \left( \frac{v_i}{v_0} \right)^\delta - \left( \frac{s^*(v_i, \Delta v_i)}{s_i} \right)^2 \right)$$

$$s^*(v_i, \Delta v_i) = s_0 + v_i T + \frac{v_i \Delta v_i}{2\sqrt{u_i^{\max} \cdot b}}$$

$$s_i = p_{i-1} - p_i - l_{i-1}$$

$$\Delta v_i = v_i - v_{i-1} \quad (1)$$

where  $i$  is the vehicle being considered,  $i - 1$  is the vehicle in front and so on;  $u_i, v_i, p_i, l_i$  is the acceleration, velocity, location and length of vehicle  $i$ ;  $s_i$  is the actual distance between vehicles  $i$  and  $i - 1$ ;  $u_i^{\max}$  is the maximum possible acceleration of the vehicle.  $b$  represents comfortable braking strength. The desired velocity and minimum distance between vehicles is denoted by  $v_0$  and  $s_0$  respectively.  $T$  represents the time headway observed by the vehicle;  $\Delta v_i$  is the difference

in the velocities of vehicle  $i$  and vehicle  $i - 1$  in front;  $\delta$  corresponds to a factor which can be tuned to control the behavior of the vehicle. Bigger the value of  $\delta$ , more aggressive the reaction of the vehicle in general.

### C. Model Mismatch

The centralized controller needs to use the future behavior (controls) of MDVs to compute controls for CACC vehicles. The future controls of MDVs is not known, it can only be predicted. To predict future controls, we assume simple but realistic MDV models. This leads to a model mismatch between the (true) MDV control and the predicted control (derived from the simplified MDV model) which must be considered.

## IV. PROPOSED CONTROLLER

In this section, we describe the proposed controller. We first detail two possible assumed models of the MDV used to predict future MDV controls within the centralized controller. Secondly, we propose a method for extending the size of vehicles countering localization error. Then, we describe our proposed control strategy, followed by the buffer implementation to ensure control availability in case of infeasibility.

### A. MDV braking model assumed by centralized controller

At each time slot  $n$  in the simulation horizon  $N_s$  the centralized controller uses a particular control (braking) model for MDV to predict braking control values  $u_{i,\eta}$  for MDV over the entire prediction horizon  $N_p$  (i.e.,  $\eta = 1, \dots, N_p$ ) after current time  $n$ .  $v_{i,\eta+n}$  corresponds to the predicted velocity of vehicle  $i$  at the  $\eta^{th}$  time slot in the future, computed at time slot  $n$ . Two models used in the simulations are introduced next.

1) *Model 1*: This profile assumes MDVs begin to brake after an assumed perception response time  $t_{i,1}$  after the vehicle in front starts to brake, and they continue to brake with a fixed magnitude  $u_i^{\min}$  until halt (zero velocity). Thus,

$$u_{i,\eta} = \begin{cases} 0 & \eta + n \leq ct_{i,1} \\ u_i^{\min} & v_{i,\eta+n} \geq 0, \eta + n > ct_{i,1} \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

for all MDVs where  $u_i^{\min}$  is the maximum braking capacity.

2) *Model 2*: Model 2 is more realistic than Model 1 and can be described as follows. As before,  $t_{i,1}$  is the perception response time;  $u_i(n)$  is the actual value of the applied acceleration; the jerk  $\Delta u_i(n)$  is given by  $\Delta u_i(n) = u_i(n) - u_i(n-1)$ ;  $\Delta u_i^{\min}$  is the maximum permitted decrease in acceleration between two time slots.

Until perception response time  $n \leq ct_{i,1}$ , the controller assumes that the driver will start braking after a certain perception response time  $t_{i,1}$  and increase the braking strength gradually until it reaches a maximum. At maximum braking strength, the vehicle continues to brake until halt. Hence, when  $n \leq ct_{i,1}$ , we set

$$u_{i,\eta} = \begin{cases} 0 & \eta + n \leq ct_{i,1} \\ \max(\eta \Delta u_i^{\min}, u_i^{\min}) & v_{i,\eta+n} \geq 0, \eta + n > ct_{i,1} \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

After perception response time,  $n > ct_{i,1}$ , we discern between three cases, based on the braking magnitude:

- 1) If the braking magnitude is zero (i.e.,  $u_i(n) = 0$ ), the controller assumes that the driver will start braking at this time slot and continue to increase its braking strength gradually until the vehicle attains maximum braking strength. At maximum braking strength, the vehicle will continue to brake until halt:

$$u_{i,\eta} = \begin{cases} \max((\eta - ct_{i,1})\Delta u_i^{\min}, u_i^{\min}) & v_{i,\eta+n} \geq 0 \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

- 2) If the braking magnitude is increasing (i.e.,  $\Delta u_i(n) < 0$ ), the controller assumes that the vehicle will continue to increase its braking strength until the vehicle attains maximum braking strength. At maximum braking strength, the vehicle will continue to brake until halt:

$$u_{i,\eta} = \begin{cases} \max(u_i(n-1) + \eta c \Delta u_i(n), u_i^{\min}) & v_{i,\eta+n} \geq 0 \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

- 3) If the braking magnitude is decreasing or constant (i.e.,  $\Delta u_i(n) \geq 0$ ), the controller assumes, the vehicle will continue to brake at the previous braking magnitude until halt:

$$u_{i,\eta} = \begin{cases} u_i(n-1) & v_{i,\eta+n} \geq 0 \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

## B. Dealing with Localization Errors

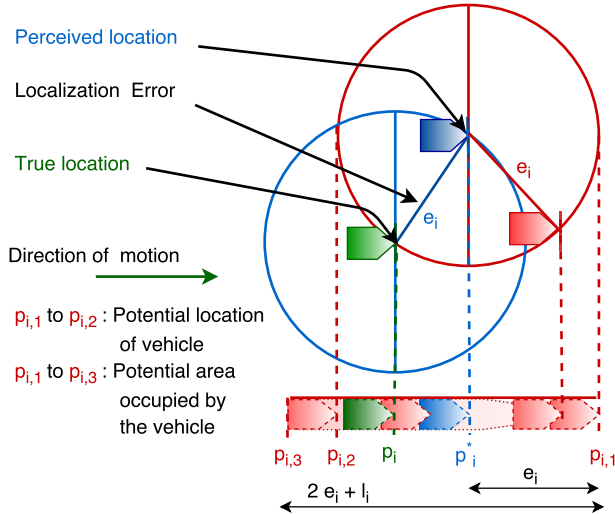


Fig. 2. Converting a 2D perceived localization information into a 1D localization information in presence of localization errors.

Fig. 2 depicts our modeling of localization errors. In this figure, vehicle  $i$  has a perceived location  $p_i^*$  different from its true location  $p_i$  by a localization error of magnitude  $e_i$ . The perceived and true location of the vehicle is denoted by blue and green blocks respectively. The vehicle can be located anywhere within a circle centered at the perceived location  $p_i^*$  with radius equal to  $e_i$ . As a vehicle position is

usually computed in 2D, the localization error is also in 2D. But without loss of generality, we only consider longitudinal motion of vehicles on a single lane to simplify our study. Thus, we transform this 2D scenario to a 1D scenario (as shown in the bottom part of the Fig. 2). The vehicle can be located anywhere between  $p_{i,1}$  and  $p_{i,2}$ .

We assume that a vehicle's position refers to its front bumper and that the true occupied road length corresponds to the vehicular length  $l_i$  from the front to the rear bumper. The potential length of the road occupied, where the vehicle may be located lies accordingly between  $p_{i,1}$  to  $p_{i,3}$ . As we can not be certain about the position of the vehicle (which may be anywhere between  $p_{i,1}$  to  $p_{i,2}$ ) and the corresponding road occupancy between  $p_{i,1}$  to  $p_{i,3}$ , in the proposed approach, the vehicle is *assumed* to be at  $p_{i,1}$  and the new length of the vehicle is *assumed* to be  $l_{i,e}$ . Thus each vehicle is modeled as having a length proportional to the localization uncertainty. (7a), and (7b) express the above mentioned ideas mathematically.

$$l_{i,e} = l_i + 2e_i \quad (7a)$$

$$p_{i,1} = p_i^* + e_i \quad (7b)$$

The assumed distance between vehicles  $i$  and  $k$  can be represented using  $d_{i,k}^*$ .  $d_{i,k}^*$  is the distance between the assumed locations of vehicles  $i$  and  $k$  and is mathematically represented in (8). Kindly refer to [15] for further details. Note that neither transmitting nor receiving entities (vehicles or centralized controllers) are aware of vehicles' true locations.

$$d_{i,k}^*(n) = p_{i,1}(n) - p_{k,1}(n) - l_{i,e} \quad i \in 2 \dots n_v \quad (8)$$

Equations (7a), (7b) and (8) represent the proposed method to account for localization errors.

## C. Controller Formulation

The control system knows neither the true positions nor the true distance between vehicles. It computes control inputs using perceived positions and perceived distance between vehicles. The centralized controller is assumed to have full knowledge (at instant  $n = 0$ ) about the state parameters (perceived position and velocity) of all MDVs and CACC vehicles, including their vehicular constraints. Our proposal is to calculate control inputs for CACC vehicles taking into account CACC vehicles and MDVs at each instant  $n$  over the simulation horizon  $N_s$  ( $n = 1, \dots, N$ ). We model our centralized controller as an MPC, as described below.

1) *Dynamics and Constraints*: The state variable  $x_i$  of a vehicle  $i$  ( $i \in 1 \dots n_v$ ) is defined as the perceived position  $p_{i,1}$ , velocity  $v_i$  tuple in (9).

$$x_i = [p_{i,1} \ v_i]^T \quad (9)$$

The general kinematic relation between position, velocity, acceleration and jerk is given by (10). We assume acceleration to remain constant between two time slots.

$$\begin{aligned} u_i(n+1) &= \Delta u_i(n+1) + u_i(n) \\ v_i(n+1) &= v_i(n) + u_i(n)\Delta t \end{aligned} \quad (10)$$

$$p_i(n+1) = p_i(n) + v_i(n)\Delta t + 0.5u_i(n)(\Delta t)^2$$

A discrete time linear control system represented by (11) is used, where values for constants are given by (12),

$$x_i(n+1) = Ax_i(n) + Bu_i(n) \quad (11)$$

$$A = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \quad B = \begin{bmatrix} (\Delta t)^2/2 \\ \Delta t \end{bmatrix} \quad (12)$$

where  $\Delta t$  is the time between two consecutive time slots  $n$  and  $n+1$ . Vehicle and road constraints in terms of minimum and maximum values of position, velocity, acceleration are accounted for in (13a), and (13b),

$$\begin{bmatrix} p_{i,1}^{\min} \\ v_i^{\min} \end{bmatrix} \leq x_i(n) \leq \begin{bmatrix} p_{i,1}^{\max} \\ v_i^{\max} \end{bmatrix} \quad (13a)$$

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

where  $(\cdot)_i^{\min}$ ,  $(\cdot)_i^{\max}$  corresponds to minimum and maximum value of that parameter for vehicle  $i$ .  $u_i^{\min}$  and  $u_i^{\max}$  stand for maximum braking and maximum acceleration capabilities. Restricting jerks  $\Delta u$  within certain acceptable bounds ensures smooth braking for CACC vehicles and is implemented using (14). Note: MDVs implement a braking profile defined by model in Section III-B and thus jerks corresponding to MDVs can not be optimized.

$$\Delta u_i^{\min} \leq \Delta u_i(n) \leq \Delta u_i^{\max} \quad (14)$$

Collision avoidance for CACC vehicles is achieved by ensuring the perceived distance between vehicles is always positive. If  $Z$  be the set of all CACC vehicles amongst  $n_v$  vehicles, this condition can be given as:

- Front-end collision avoidance:

$$d_{i,k}^*(n) > 0 \quad i \in Z, \quad k = i - 1 \quad (15)$$

- Front and rear-end collision avoidance

$$\begin{aligned} d_{i,k}^*(n) > 0 & \quad i \in Z, \quad i > 1, \quad k = i - 1 \\ d_{i,k}^*(n) > 0 & \quad i \in Z, \quad i < n_v, \quad k = i + 1 \end{aligned} \quad (16)$$

If the control profile of MDV models the relation between a MDV and a CACC vehicle, the centralized controller could be able to avoid accidents on both, MDV and CACC vehicles, but this is out of scope of this paper. Starting and terminal position and velocity can be represented as constants  $x_i(0)$  and  $x_i(N_s)$ .  $p_i(0)$  and  $p_i(N_s)$  indirectly defines the range of the vehicle and the path it needs to follow in a 1D scenario. (17) finally ensures the terminal velocity of all vehicles reach zero and this signifies a braking scenario.

$$v_i(N_s) = 0 \quad (17)$$

2) *MPC Controller Description*: Comfort is related to the change in control inputs (acceleration). Strong deviations in control inputs can be penalized using the 2-norm. To present the cost function in a quadratic form, we use the square of the 2-norm of change in control inputs between two time slots. By integrating all of previously defined, the optimization problem

for a centralized mixed vehicle braking coordination scenario can be represented as:

$$\text{minimize} \quad \sum_{i=1}^{n_v} \sum_{\eta=1}^{N_p} (u_i(\eta) - u_i(\eta-1))^2 \quad (18)$$

subject to

MDV model, (7a), (7b), (9), (10),  
(11), (12), (13a), (13b), (14), (16), (17)

Assumed MDV model generates predicted acceleration for MDVs which are set as constraints in the above MPC problem. At each time slot, state parameters, predicted MDV controls and constraints are updated and the centralized controller solves the convex optimization problem represented by (18). In this paper, we rely on QUADPROG toolbox in MATLAB to solve (18).

#### D. Infeasibility: Control Buffer Implementation

At each time slot, localization errors are modeled in a robust manner to avoid collisions and they are used by the centralized controller to compute control inputs for CACC vehicles based on the updated state information of all (considered) vehicles using (18). The output of (18) is a vector of control actions for each CACC vehicle, over a *prediction horizon*. This control vector is sent in downlink to CACC vehicles. As soon as control inputs are obtained, the first value of control is applied and the rest are stored in a buffer. The buffer is updated at each reception of new control inputs.

At certain time slots either due to the uncertainty in localization errors or model mismatch or the robust-modeling of localization errors or violation of the collision avoidance constraint, the feasible set of the optimization problem could be empty and the control problem might be infeasible. When computation is infeasible and no control data is transmitted in downlink, control data corresponding to that time slot from the previous computation stored in the buffer is used.

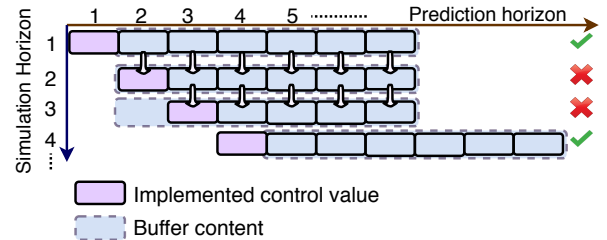


Fig. 3. Buffer implementation: The use of control data stored in the buffer when new control data is not received is illustrated.

The implementation of the buffer is illustrated in Fig. 3. Assume at the first slot, controls are successfully received at the vehicle. The first control value is implemented (shaded purple) and the rest are stored in the buffer (shaded blue). At the next simulation slot, assume new controls are not received; the buffer content is retained and the first control value from the buffer is used by the vehicle and that value is then deleted. At the third time slot, assume new controls are not received, buffer is retained, the next control value is applied and then



that value is deleted. At the fourth time slot, assume new data is received, first value is applied and the buffer content is replaced with the rest of the new control data.

If the optimization problem is infeasible at the beginning, the algorithm ignores the jerk filter to allow any limit of jerk for the first time slot (within maximum braking capacity). If even then, the optimization problem is infeasible, CACC vehicles brake at the maximum braking capacity permitted by the jerk filter (e.g.: If permitted jerk is  $2.5 \text{ m/s}^3$  and the acceleration in the previous time slot was zero, the permitted braking strength for the current time slot would be  $-0.25 \text{ m/s}^2$ ). The optimization problem is retried at the next time slot with updated parameter values.

## V. SIMULATIONS AND ANALYSIS

The goal of these simulations is to highlight the impact of the uncertainty created due to model mismatch between the assumed and the actual braking model behavior of MDVs and constantly changing localization error in a centralized control scenario. Other errors that impact a centralized controller operation like communication and lower-level controller errors are out of scope of this paper.

First, we highlight the impact of model mismatch on a centralized controller assuming perfect localization. We then compare the impact of model mismatch in presence of localization errors where the centralized control algorithm does not account for localization errors (a non-robust controller) versus the impact on the control algorithm that accounts for localization errors (a robust controller).

### A. General simulation settings

We assume at the start of the simulation, 4 vehicles are approaching an intersection. The initial vehicle is located at a distance of 800 meters from the intersection. The first vehicle moves towards the intersection with a constant velocity until it is notified of a potential obstacle or an intersection at the *notification distance* from the obstacle. Notification distance is also the distance at which the first vehicle senses an obstacle or is notified about an obstacle. An's exemplary study illustrated that at least one DSRC/ITS-G5 safety message would be received with 99.5% probability by the time notification distance reduces to 95.9 m [26]. But in this work, a range of *notification distances* from 95.9 m to 150 m are used for simulations. All following vehicles have a zero initial acceleration, and then are governed by IDM until the notification distance. We assume the stream of vehicles will reach steady state by the time first vehicle is notified. After the first vehicle is notified, MDV and CACC vehicles assume corresponding braking models. A centralized controller intervenes and assists a collision free braking procedure.

Equal number of CACC and MDV vehicles (2 each) are used. There are six possible arrangements of these 4 vehicles and 20 samples of each arrangement are taken to create a database of 120 simulation samples. We define a simulation *sample* as the set of vehicle state parameters. Each sample has different initial velocities ( $90 \text{ kmph} \pm 5\%$ ) and different initial distance between vehicles. We assume vehicles can

TABLE I  
GENERAL PARAMETERS

Parameter	Value	Parameter	Value
$g$	$9.88 \text{ m/s}^2$	$(\Delta u^{\min}, \Delta u^{\max})$	$(-0.25, +0.25)$
$l_i$	4 m	$\delta$	4
$\Delta t$	0.1 s	$b$	$-2 \text{ m/s}^2$
$u^{\min}$	$-5.928 \text{ m/s}^2$	$N_s, N_p, N_e$	100
$u^{\max}$	$1 \text{ m/s}^2$	$(v^{\min}, v^{\max})$	$(0, \infty)$
$(p^{\min}, p^{\max})$	$(0, \infty)$		

only have non-negative velocity signifying, they can not move in reverse. The obstacle or the intersection is assumed to be at the origin. A simulation is counted to be successful when the terminal velocity of all vehicles reach zero and is considered a failure if there are any collisions. The perception response time of a MDV is drawn from a normal distribution  $\mathcal{N}(1.33, (0.27)^2)$  [27] and is capped between 0.8 s and 1.8 s. Jerk value around  $2 \text{ m/s}^3$  is usually considered comfortable in the literature [28]. In this work, increase and decrease in braking capacity is restricted to  $2.5 \text{ m/s}^3$  and  $-2.5 \text{ m/s}^3$  respectively. As there are 10 time slots per second,  $\Delta u^{\min}, \Delta u^{\max}$  values are  $(-0.25, +0.25)$  respectively.

Localization errors are derived from a zero-mean Gaussian distribution with standard deviation  $\phi$ . The actual localization error  $e_i$  is drawn for each vehicle  $i$  from the Gaussian distribution. In a *heterogeneous simulation scenario* different vehicles use different localization techniques, we assume MDVs use GPS/GNSS based localization techniques, with a std of error of 4 m [2], whereas CACC vehicles use advanced localization techniques like map matching to compute localization with a std of error of 25 cm. In a *homogeneous simulation scenario*, we assume all vehicles have the same localization technique and thus share the same  $\phi$  value. The controller assumes  $e_i$  remains constant for each vehicle over the prediction horizon but changes at every time slot in simulation horizon. Further parameters are shown on Table I.

We analyze simulations by observing the percentage of collisions avoided (CA) in this braking scenario. At every successful computation, although the buffer is updated, it is not used. We consider the buffer to be used only in case of a computational failure. The total number of collisions avoided (CA) is the sum of the number of collisions avoided without the use of buffer (CAWOB) and number of collisions avoided with the use of buffer (CAWB),  $CA = CAWOB + CAWB$ .

### B. Simulation set 1

In this particular study, the goal is to highlight the impact of model mismatch on a centralized controller in the absence of localization error. Different assumed models give rise to different model mismatches. Two kinds of model mismatches are generated: first by using the profile introduced in Section IV-A1 as the assumed MDV model whereas the profile introduced in Section III-B as the actual MDV model; second by using the profile introduced in Section IV-A2 as the assumed MDV model whereas the profile introduced in Section III-B as the actual MDV model. We propose the use of a *Control Buffer* introduced in Section IV-D.

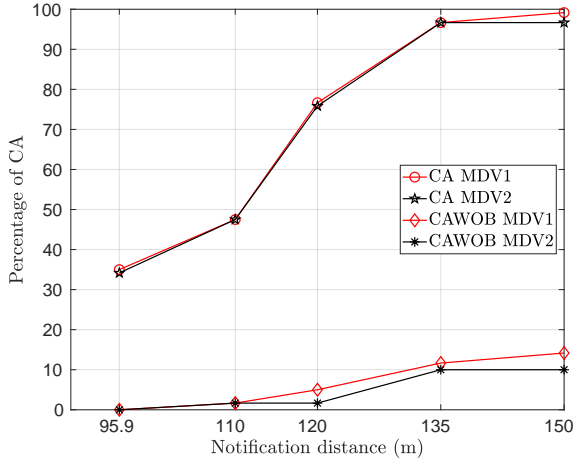


Fig. 4. Collision avoidance results evaluating model mismatch, but no localization error: most of the CA are attributed to the use of the buffer.

The collision free braking control computation shall be infeasible when the assumed MDV profile leads to unreasonable constraints. The use of a buffer facilitates the availability of control inputs despite computational infeasibility and the use of receding horizon MPC with small time slots (0.1s) facilitates frequent recomputation of controls which mitigates the impact of model mismatch. If we consider simulation results over entire database (refer Fig. 4), we observe that the percentage of CA without the use of buffer (CAWOB) is very small compared to the total collisions avoided (CA) irrespective of the assumed MDV profile. As  $CA = CAWOB + CAWB$ , it follows that the other collisions are avoided with the use of the buffer (CAWB) (not plotted). It is evident from Fig. 4 that the use of buffer leads to a higher number of collisions avoided ( $CAWB > CAWOB$ ) and thus the use of buffer is required.

Figure 4 also shows that the percentage of collisions avoided under different model mismatch scenarios could be different. This is because different assumed MDV profiles (different types of model mismatch) lead to different constraints which could lead to a different number of total collisions avoided as seen for notification distance 150 m in Fig. 4. To further illustrate the impact of different assumed MDV models (and model mismatches) on controls, we demonstrate two samples with the same set of initial parameters (see Figs. 5–6). The order of vehicles is [MDV;CACC;CACC;MDV], first MDV vehicle is the leading vehicle. The acceleration profile of vehicles in both scenarios is the same until the notification distance (in this case, 110 m) as all vehicles are assumed to implement IDM. Once they enter the notification distance, CACC vehicles use controls transmitted by the centralized controller whereas MDVs react to the vehicle in front. As the controls computed by the centralized controller are different for different assumed MDV control models, the acceleration of CACC vehicles is different and the acceleration (reaction) of MDVs is also different, as evident from Fig. 5 and Fig. 6.

Different controls lead to a different amount of discomfort. Discomfort is computed as the two norm of change

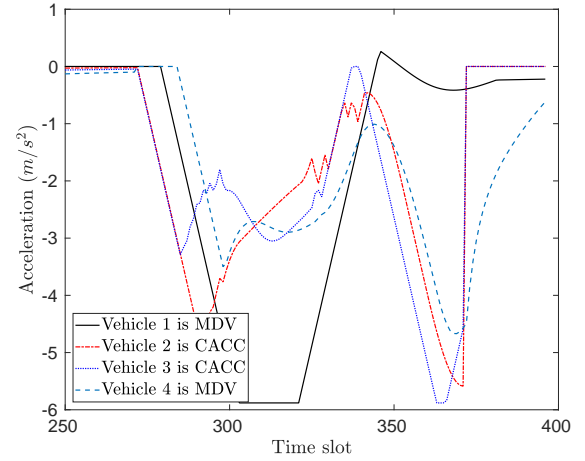


Fig. 5. Acceleration profile for a particular simulation with no localization error and MDV profile 1.

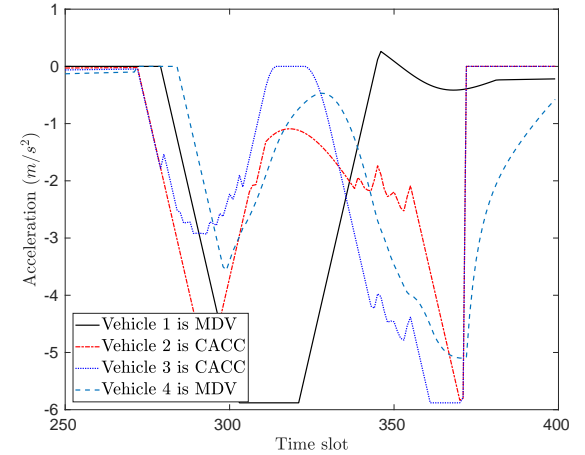


Fig. 6. Acceleration profile for a particular simulation with no localization error and MDV profile 2.

in acceleration (per time slot) over the entire simulation.<sup>1</sup> The average value of discomfort per vehicle for simulations where collisions were avoided is plotted in Fig. 7. Generally speaking, we observe that the assumed MDV profile 2 results into lower levels of discomfort as it is more realistic compared to the assumed MDV profile 1.<sup>2</sup> In this paper, control values from buffer are only used when control computation is infeasible. We observe that the average discomfort in cases where collisions were avoided when controls from buffer are used (CAWB) is higher than the discomfort value when buffer is not used (CAWOB) in Fig. 7. The use of a buffer results into more collision avoidance at the cost of a higher discomfort. As all simulations without the use of the buffer at notification distance of 95.9 m resulted into collisions, there is no value

<sup>1</sup>The two norm of change in acceleration is also proportional to fuel consumption

<sup>2</sup>The reason for the discomfort value using MDV model 2 at the notification distance of 120 m being higher than that of assumed MDV model 1 could not be determined.



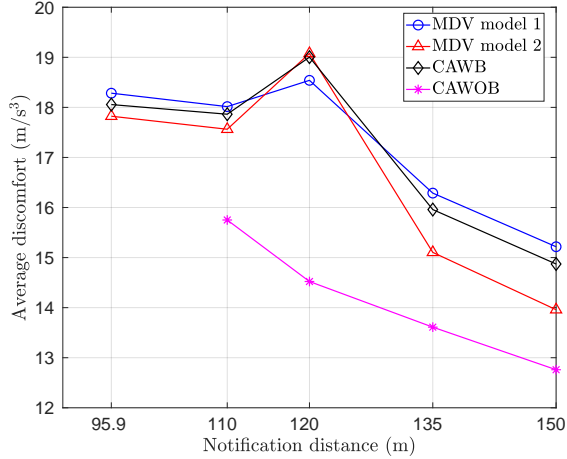


Fig. 7. Average values of discomfort in cases where collisions were avoided for different values of notification distance (no localization error).

of discomfort in the plot.

### C. Simulation set 2

In this particular study, the goal is to highlight the impact of model mismatch on a centralized controller in presence of localization error, when the algorithm doesn't account for localization errors (a non-robust MPC controller). We choose to use model introduced in Section IV-A2 as the assumed MDV model. Buffer is used to counter infeasibilities. When localization errors are not accounted for, control inputs are generated using perceived information (positioning information computed by the vehicle) which is different from the actual localization. Due to the difference in the true position and the perceived information, control inputs generated are usually different, which could be a source of collisions. Control inputs generated using perceived information are applied on the vehicles in their true positions.

First we consider a homogeneous localization system where all vehicles have the same  $\phi$ . Results are displayed in from of a bar plot in Fig. 8. To analyze the figure, choose a value of notification distance and observe the change in CA as  $\phi$  increases. As localization errors are not accounted for, in general, CA decreases as the value of  $\phi$  increases<sup>3</sup>. Next, consider a stream of vehicles with heterogeneous localization system, where std of localization errors is different for different vehicles; for CACC vehicles,  $\phi = 4$  m, for MDV  $\phi = 0.25$  m. Simulation results reveal that total CA (sum of collisions avoided with and without the use of buffer) increases with an increase in notification distance as evident from the non-robust MPC plot in Fig. 10.

### D. Simulation set 3

In this particular study, the goal is to highlight the impact of model mismatch on a centralized controller in presence of

<sup>3</sup>The simulations involve several sources of randomness. Hence, due to the limited number of scenarios, the value of CA in Fig. 8 is not strictly decreasing with the increase in standard deviation  $\phi$  of localization error.

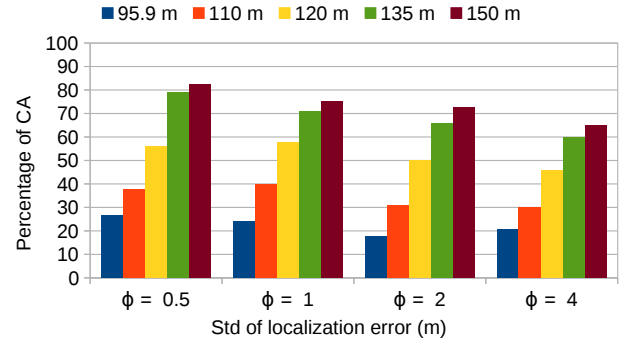


Fig. 8. Evaluating a non-robust centralized controller in presence of localization errors in a system consisting of vehicles with the same localization system (a homogeneous system). The legend mentions different values of notification distance.

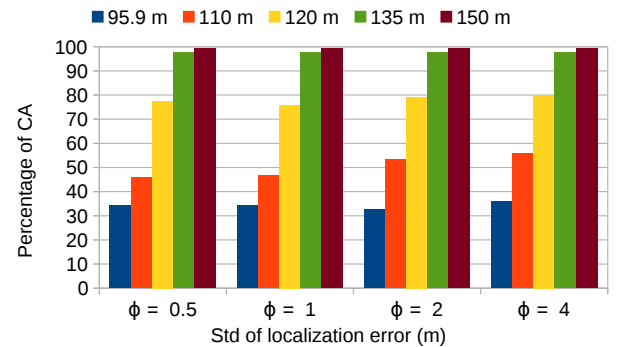


Fig. 9. Evaluating a robust centralized controller in presence of localization errors in a system consisting of vehicles with the same localization system. The legend mentions different values of notification distance.

localization error, using a robust MPC where the algorithm accounts for localization errors. Buffer is used to counter infeasibilities.

First, let us consider simulations consisting of vehicles with homogeneous localization systems (and thus the same std of localization error). Simulation results are plotted in Fig. 9. For a notification distance of 150 m, we observe that CA nears 100% despite the presence of localization errors ( $\phi \in 0.5, 1, 2, 4$ ). On comparing Fig. 8 and Fig. 9 for any particular value of  $\phi$  and notification distance, we observe that the relative performance of the robust algorithm is much better compared to that of the non-robust algorithm. Next, consider a stream of vehicles with heterogeneous localization system. Simulation results plotted in Fig. 10 show that for a notification distance of 135 m and more, we can attain almost 100% CA using a robust controller instead of approximately 80% with a non-robust controller. Comparing the two plots in Fig. 10, we observe the percentage of CA with a robust MPC is higher than that of a non-robust case, for all notification distances. This highlights the benefits of implementing a robust MPC controller; it avoids more collisions compared to a non-robust controller by effectively mitigating model mismatches and localization errors.

On analyzing Fig. 9 in detail, we observe that despite an increase in  $\phi$ , CA increases. e.g.: at the notification distance

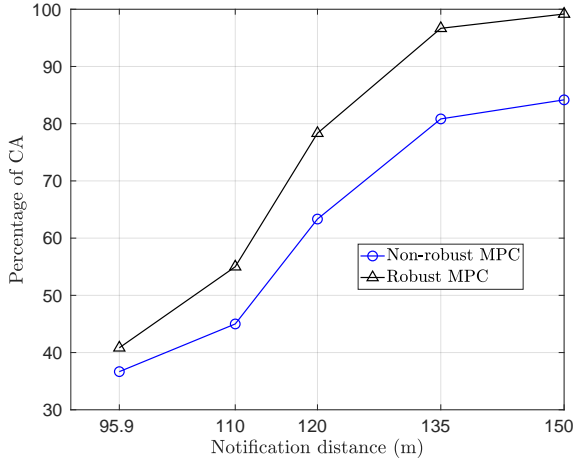


Fig. 10. Comparing the performance of a non-robust and a robust centralized controller in presence of localization errors in a system consisting of vehicles with different localization systems (a heterogeneous system).

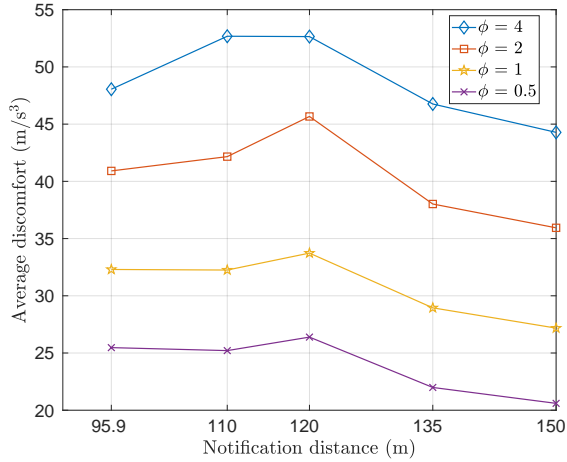


Fig. 11. Average values of discomfort in cases where CA was successful for different values of  $\phi$  and notification distance.

of 110 m, percentage of CA for  $\phi = 1$  m and  $\phi = 4$  m is 46.66 and 55.8 respectively. This is because the algorithm assumes vehicles to be located over a larger area and thus offers a larger inter-vehicular distance when localization error is larger. Larger inter-vehicular distance provides extra margin and can indirectly compensate for model mismatch as well.

The assumed vehicle length and the area over which the vehicle is assumed to be located evolves with every change in the localization error. This results into drastic changes in computed controls. Bigger the change in localization error, bigger the change in acceleration. We compute discomfort for cases where collisions were avoided and plot it in Fig. 11. We observe that discomfort is higher for higher values of std of localization error. From Fig. 11 and Fig. 9 we can conclude that for higher values of standard deviation of localization error, the proposed algorithm can avoid more collisions, at the cost of an increased discomfort, despite model mismatches.

## VI. CONCLUSIONS

In this work, the impact of localization errors and model mismatch on a centralized robust Model Predictive Controller (MPC) has been analyzed. Model mismatch arises when the assumed and the actual model of vehicle is different. Localization errors arise due to inaccurate localization. A multi-vehicle braking scenario consisting of vehicles with different levels of automation and different standard deviation of localization errors is simulated. Different model mismatches usually leads to different controls of vehicles which leads to different levels of discomfort and different results (different number of collisions). In case of computational infeasibility arising due to localization errors or model mismatch, controls from the buffer can be used. The implementation of a receding horizon MPC ensures control recomputation at every time slot which is essential to counter model mismatch and localization errors. Simulations show that the robust MPC controller implemented with a buffer results into more collisions avoided compared to the non-robust controller at the cost of increased discomfort. Bigger the localization error, higher is the number of collisions avoided by robust controller compared to non-robust controller. Larger the notification distance, more is the number of collisions avoided in general.

In future, the proposed robust centralized MPC controller implementing a control buffer will be made more realistic by implementing engine delay and the performance of the controller will be analyzed under communication uncertainties.

## REFERENCES

- [1] Qing Xu and Raja Sengupta. Simulation, analysis, and comparison of ACC and CACC in highway merging control. In *Intelligent Vehicles Symposium, 2003. Proceedings. IEEE*, pages 237–242. IEEE, 2003.
- [2] US DoD. Global positioning system standard positioning service performance standard. *Assistant secretary of defense for command, control, communications, and intelligence*, 2008.
- [3] Jesse Levinson, Jake Askeland, Jan Becker, Jennifer Dolson, David Held, Soeren Kammel, J Zico Kolter, Dirk Langer, Oliver Pink, Vaughan Pratt, et al. Towards fully autonomous driving: Systems and algorithms. In *Intelligent Vehicles Symposium (IV), 2011 IEEE*, pages 163–168. IEEE, 2011.
- [4] HIGH PRECISION POSITIONING FOR COOPERATIVE-ITS. <http://highlights.eu/> [Online; accessed 4-April-2017].
- [5] Raj Hareesh Patel, Jérôme Härrli, and Christian Bonnet. A collision mitigation strategy for intelligent vehicles to compensate for human factors affecting manually driven vehicles. In *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, pages 114–119, Oct 2017.
- [6] I Ge Jin and Gábor Orosz. Data-driven parameter estimation for optimal connected cruise control. In *Decision and Control (CDC), 2017 IEEE 56th Annual Conference on*, pages 3739–3744. IEEE, 2017.
- [7] J. Monteil, N. OHara, V. Cahill, and M. Bourouche. Real-Time Estimation of Drivers' Behaviour. In *2015 IEEE 18th International Conference on Intelligent Transportation Systems*, pages 2046–2052, Sept 2015.
- [8] Alexis C Madrigal. The trick that makes google's self-driving cars work. *The Atlantic*, May, 2014. [Online; accessed 30-June-2016].
- [9] Na Chen, Meng Wang, Tom Alkim, and Bart van Arem. A Robust Longitudinal Control Strategy of Platoons under Model Uncertainties and Time Delays. *Journal of Advanced Transportation*, 2018, 2018.
- [10] Wouter J Schakel, Bart Van Arem, and Bart D Netten. Effects of cooperative adaptive cruise control on traffic flow stability. In *Intelligent Transportation Systems (ITSC), 2010 13th International IEEE Conference on*, pages 759–764. IEEE, 2010.
- [11] Feng Gao, Shengbo Eben Li, Yang Zheng, and Dongsuk Kum. Robust control of heterogeneous vehicular platoon with uncertain dynamics and communication delay. *IET Intelligent Transport Systems*, 10(7):503–513, 2016.

- [12] Maytheewat Aramrattana, Raj Haresh Patel, Cristofer Englund, Jérôme Härrri, Jonas Jansson, and Christian Bonnet. Evaluating Model Mismatch Impacting CACC Controllers in Mixed Traffic using a Driving Simulator. *Intelligent Vehicles Symposium (IV)*, IEEE, 2018. available at: <http://www.eurecom.fr/fr/people/patel-raj-haresh-1/publications>.
- [13] M. Mazzola, G. Schaaf, A. Stamm, and T. Kürner. Safety-Critical Driver Assistance Over LTE: Toward Centralized ACC. *IEEE Transactions on Vehicular Technology*, 65(12):9471–9478, Dec 2016.
- [14] Carlos M Massera, Marco H Terra, and Denis F Wolf. Safely optimizing highway traffic with robust model predictive control-based cooperative adaptive cruise control. *arXiv preprint arXiv:1605.07493*, 2016.
- [15] Raj Haresh Patel, Jérôme Härrri, and Christian Bonnet. Accounting for localization errors in a mixed-vehicle centralized control system. In *Mobil TUM 2017, International Scientific Conference on Mobility and Transport, July 4-5, 2017, Munich, Germany*, 06 2017. Available at <http://www.eurecom.fr/fr/people/patel-raj-haresh-1/publications>.
- [16] Raj Haresh Patel, Jérôme Härrri, and Christian Bonnet. Impact of localization errors on automated vehicle control strategies. In *2017 IEEE Vehicular Networking Conference (VNC)*, pages 61–68, Nov 2017.
- [17] Georg Schildbach, Matthias Soppert, and Francesco Borrelli. A collision avoidance system at intersections using Robust Model Predictive Control. In *Intelligent Vehicles Symposium (IV)*, 2016 IEEE, pages 233–238. IEEE, 2016.
- [18] G. Rodrigues de Campos, P. Falcone, R. Hult, H. Wymeersch, and J. Sjöberg. Traffic coordination at road intersections: Autonomous decision-making algorithms using model-based heuristics. *IEEE Intelligent Transportation Systems Magazine*, 9(1):8–21, Spring 2017.
- [19] Jackeline Rios-Torres and Andreas 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*, 2016.
- [20] Xiao-Yun Lu, Jianqiang Wang, Shengbo Eben Li, and Yang Zheng. Multiple-vehicle longitudinal collision mitigation by coordinated brake control. *Mathematical Problems in Engineering*, 2014, 2014.
- [21] Jian-Qiang Wang, Shengbo Eben Li, Yang Zheng, and Xiao-Yun Lu. Longitudinal collision mitigation via coordinated braking of multiple vehicles using model predictive control. *Integrated Computer-Aided Engineering*, 22(2):171–185, 2015.
- [22] Ellen van Nunen and Jan Verhaegh. Robust model predictive cooperative adaptive cruise control subject to V2V impairments. In *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2017.
- [23] The European Commission. 5G-PPP White Paper on Automotive Vertical Sectors. <https://5g-ppp.eu/wp-content/uploads/2014/02/5G-PPP-White-Paper-on-Automotive-Vertical-Sectors.pdf>, 2015. [Online; accessed 26-May-2018].
- [24] Arne Kesting, Martin Treiber, and Dirk Helbing. Enhanced intelligent driver model to access the impact of driving strategies on traffic capacity. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 368(1928):4585–4605, 2010.
- [25] Martin Treiber, Arne Kesting, and Dirk Helbing. Delays, inaccuracies and anticipation in microscopic traffic models. *Physica A: Statistical Mechanics and its Applications*, 360(1):71–88, 2006.
- [26] N. An, T. Gaugel, and H. Hartenstein. Vanet: Is 95% probability of packet reception safe? In *2011 11th International Conference on ITS Telecommunications*, pages 113–119, Aug 2011.
- [27] Shane B McLaughlin, Jonathan M Hankey, Thomas A Dingus, and Sheila G Klauer. Development of an few algorithm evaluation methodology with evaluation of three alert algorithms. *National Highway Traffic Safety Administration, Tech. Rep.*, 2009.
- [28] ISO2631-1. Mechanical vibration and shock - evaluation of human exposure to whole-body vibration - Part 1: General requirements. *International Organization for Standardization*, 1997.



**Raj Haresh Patel** is a final year doctoral student at EURECOM in France. His main research interests lie in collision avoidance between multiple vehicles with different levels of automation. He currently focuses on mixed vehicle scenarios with autonomous vehicles and manually driven vehicles. He has a vision for future transportation system where a centralized entity monitors vehicles and intervenes when necessary to ensure a safer driving experience.



**Henk Wymeersch** received the Ph.D. degree in Electrical Engineering/Applied Sciences in 2005 from Ghent University, Belgium. He is currently an Associate Professor with the Department of Signals and Systems at Chalmers University of Technology, Sweden. Prior to joining Chalmers, he was a Post-doctoral Associate with the Laboratory for Information and Decision Systems (LIDS) at the Massachusetts Institute of Technology (MIT). He served as Associate Editor for IEEE COMMUNICATION LETTERS (2009–2013), IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS (2013–present), and the TRANSACTIONS ON EMERGING TELECOMMUNICATIONS TECHNOLOGIES (2011–present).



**Jérôme Härrri** is an Associate Professor with the Communication Systems Department, EURECOM, Sophia Antipolis, France, where he leads the Connected Automated Transport System (CATS) team in the Networked Systems group. Previously, he led the Traffic Telematics Junior Research Group of the Institute of Telematics, Karlsruhe Institute of Technology, Karlsruhe, Germany. His research interests are related to wireless vehicular communication and networking, traffic flow modeling, positioning and localization, or control system optimization, in particular their mutual interactions in future ADAS and ITS. He has authored and coauthored over 70 international journal and conference papers, is involved in various national and European research projects related to connected vehicles, and actively contributes to various Standardization bodies. Jérôme Härrri received the M.Sc. and the Dr. ès Sc. (PhD) degrees in telecommunication from the Swiss Institute of Technology (EPFL), Lausanne, Switzerland.



**Christian Bonnet** joined EURECOM in 1992 after more than 12 years in industry. He was at the head of the Mobile Communications Department of EURECOM from 1998 to 2011. He is currently leading the Wireless Systems and Protocols research group. He participated in many (FP5, 6, 7, H2020) European projects dealing with mobility features based on IPv6 in heterogeneous mobile systems, Software defined Radio, Mesh architectures for public safety systems. His current field of research is on M2M and Internet of Things. He is leading the president of the open source OpenAirInterface Software Alliance targeting 5G systems. He is the technical coordinator of several French national projects mixing LTE access and M2M services. He is involved in the Secured Communicating Solutions regional cluster for innovation as co-leader of the Mobiles services and M2M strategic axe. He co-authored more than 200 publications in international conferences.