

Model Predictive Control Allocation of Systems with Different Dynamics

Moad Kissai, Bruno Monsuez, Xavier Mouton, Didier Martinez, Adriana Tapus

▶ **To cite this version:**

Moad Kissai, Bruno Monsuez, Xavier Mouton, Didier Martinez, Adriana Tapus. Model Predictive Control Allocation of Systems with Different Dynamics. 2019 IEEE Intelligent Transportation Systems Conference - ITSC, Oct 2019, Auckland, New Zealand. pp.4170-4177, 10.1109/ITSC.2019.8917438 . hal-02387763

HAL Id: hal-02387763
https://hal.science/hal-02387763

Submitted on 30 Nov 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Model Predictive Control Allocation of Systems with Different Dynamics

Moad Kissai[1], Bruno Monsuez[1], Xavier Mouton[2], Didier Martinez[2] and Adriana Tapus[1]

***Abstract*— Several systems are integrated in passenger cars. Some of them are just redundant systems due to safety requirements. Others, are completely different and can interact with each other as long as they are operating inside the same vehicle. Control allocation methods have been successfully implemented in advanced aircrafts to avoid conflicts, especially in the context of redundant systems. In this paper, we will rather focus on coordinating non-redundant advanced chassis systems with different dynamics. This difference in dynamics can be especially problematic when systems exhibit different communication delays. Model Predictive Control Allocation (MPCA) methods are therefore investigated in order to activate the right system at the right moment. Results show that particularly when the most effective system is saturated, another system with a different time delay can be activated few steps before saturation to instantly take over the maneuver. With good knowledge of actuator dynamics and higher computation power, MPCA methods are able to solve complex problems in severe situations.**

***Index Terms*— Model Predictive Control Allocation, Chassis Systems, Vehicle Dynamics, Identification, Robust Control.**

I. INTRODUCTION

The automotive sector is getting prepared for one of its biggest revolutions. After vehicle's electrification and connectivity, full autonomy has become the headline of many scientific researches. In order to achieve safe autonomous driving, additional sensors and actuators are added to replace the driver or at least assist him/her. Different sensors can be used in order to detect the same object. In the same way, different actuators can influence the same physical variable as the vehicle's yaw rate [1]. While the first problem is tackled by sensor fusion methods, Control Allocation (CA) algorithms are more suited for the second problem. To the best of our knowledge, these algorithms have been first introduced in the aeronautical sector [2]. Advanced aircraft systems equipped with several ailerons, rudders, elevators and so on have been coordinated using CA algorithms [2], [3], [4]. Most of the time, these algorithms are used to coordinate actuators with the same dynamics, e.g., a right aileron with a left one. Following the same idea, CA methods have been introduced in the automotive sector to manage systems with several actuators having the same dynamics. Although the actuators do have the same dynamics, they influence the vehicle differently due to their different positioning and the vehicle's non-uniform mass distribution. The Electronic Stability Program (ESP) is one of the most obvious examples in this case, as it consists of controlling the four brakes of the vehicle [5]. Another chassis system based on the same principle is the Torque Vectoring. This system attracted the automotive community as it enables controlling the vehicle's yaw rate while keeping it in the move unlike the ESP. Authors in [6] used CA algorithms to ensure Torque Vectoring control of an electric vehicle. Not only the vehicle's yaw rate control has been improved, but also the drive-train power losses have been minimized using appropriate tuning of the reference's understeer characteristics. In [7], both engine torques and brake torques have been coordinated, considering that both are based on the tire longitudinal forces. CA algorithms have been extended to systems based on different forces in [8]. A differential braking based Vehicle Dynamic Control (VDC) system and an Active Rear Steering (ARS) system have been coordinated using optimal CA algorithms. Here, the ARS system is rather based on the lateral force of the tire. Both systems have been made complementary to make a secondary different system take over the maneuver in case of failure of the primary system. Fault-tolerance has been then ensured by means of different integrated systems.

However, actuators dynamics have been most of the time ignored when designing control allocation algorithms. This could be acceptable as long as actuators with the same dynamics are involved. Once the actuators become very different, switching from one system to another can be very problematic. In [8] for example, when the ARS system fails, the VDC system takes over the maneuver instantly thanks to a modified effectiveness matrix in the CA framework that takes into account the actuators' flags. This can only be possible if both systems can be activated fast enough without any communication delays. Experiments showed us that the delay cannot be neglected. Different research works have been carried out to face time-delay issues in the control theory community. The problem has been tackled in [9] by using modified Riccati equations. In [10], delayed feedback control design for uncertain systems with time-varying input delay has been proposed. Authors of [10] used Lyapunov functions and introduced some relaxation matrices and turning parameters to make results less conservative. Results in [9] and [10] present theoretical mathematical solutions of the problem. Nonetheless, they remain tedious to implement and non adapted for industrial control applications. An alternative solution would be the Model Predictive Control Allocation (MPCA). First of all, the Model Predictive Control (MPC)

[1]Moad Kissai, Bruno Monsuez, and Adriana Tapus are with ENSTA ParisTech, Autonomous Systems and Robotics Lab, Department of Computer and System Engineering (U2IS), 828 Boulevard des Marchaux, 91762 Palaiseau Cedex, France {moad.kissai, bruno.monsuez, adriana.tapus}@ensta-paristech.fr

[2]Didier Martinez, and Xavier Mouton are with Group Renault, Chassis Systems Department, 1 Avenue du Golf, 78280 Guyancourt, France {xavier.mouton, didier.d.martinez}@renault.com

is a control method that solves an optimization problem, specifically, a quadratic program (QP), at each control interval. The solution determines the manipulated variables to be used in the plant for a number of successive intervals constituting the prediction horizon. The optimization is based on minimizing several cost functions. The most important one for reference tracking consists of minimizing the difference between the reference and the state to be controlled. The same principle can be adopted for MPCA. Here, the cost function to be minimized would be the difference between the overall effort needed to stabilize the plant and the sum of available control effectors. The delay can be taken into account in the whole prediction horizon to determine when the control request should be submitted. In this context, MPCA has been used in [11] to coordinate Hybrid Braking in electric vehicles. Results showed that the MPCA can offer faster transient response, without compromising the energy recuperation efficiency of the actuators. However, the modeling used for the different brakes is relatively simplified. In [12], MPCA has been used to stabilize a four-wheel drive electric vehicle in critical driving condition. While results showed stability improvement in severe situations, the problem exposed remains simple as the actuators considered, namely the in-wheel engines, have the same dynamics.

In this paper, we focus on completely different actuators. Our aim is to control the vehicle's yaw rate by means of the steering-based ARS system[1] and the braking-based VDC system. Moreover, as the MPC theory suffers from a lack of robustness due to its dependency on analytic modeling, here, we identify the vehicle and actuators' dynamics to be implemented in the MPCA experimentally. We also add a high-level controller based on $\mathcal{H}_\infty$ theory. Comparison with the classic CA approach shows the benefit of the MPCA in case of coordinating different systems with different dynamics, especially when both systems have to operate at the same time.

The rest of the paper is structured as follows: We start in Section II by presenting the vehicle dynamics modeling. The identification issues to be used in case of MPCA design are illustrated in Section III. In Section IV, the control allocation problem is described, along with the adaptation of the MPC theory. Comparison of classic CA methods with the MPCA are shown in Section V. The application of such complex method from an industrial point of view is discussed in VI. Conclusions and future works are outlined in Section VII.

II. VEHICLE MODELING

As its name may reveal, the MPC theory is closely based on the plant model selected. This is also the case for the MPCA. The model chosen should be as precise as possible to ensure acceptable performances in real experiments. Experimental identification seems to be the safer option. However, when carrying different experiments with different speed values, different transfer functions can be obtained. This proves the dependency of vehicle dynamics on the speed, as it was also shown in [13]. Therefore, first of all, analytic equations should be developed in order to isolate the constant parameters that need to be identified.

Another difficulty is the modular property of the CA framework. As it was discussed in [14], the multi-layered control architecture seems to be more appropriate for over-actuated vehicles. This architecture is illustrated in Fig. 1. The high-level controller in case of yaw rate control must

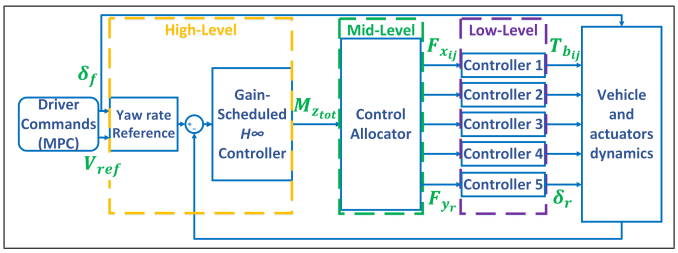


Fig. 1. Multi-layered control architecture.

generate the total yaw moment required $M_{z_{tot}}$. To synthesize this high-level controller, the following equations coming from Newton's second law of motion should be considered:

$$\begin{cases} M\left(\dot{\psi}V_x + sV_y\right) = F_{y_f}\cos\delta_{f_i} + F_{y_r}\cos\delta_{r_i} \\ \qquad\qquad + \sin\delta_{f_i}\left(F_{x_{f,l}} + F_{x_{f,r}}\right) \\ \qquad\qquad + \sin\delta_{r_i}\left(F_{x_{r,l}} + F_{x_{r,r}}\right) \end{cases} \tag{1}$$

$$\begin{cases} I_z s\dot{\psi} = F_{y_f} l_f \cos\delta_{f_i} - F_{y_r} l_r \cos\delta_{r_i} \\ \qquad + \left(l_f \sin\delta_{f_i} - \frac{t}{2}\cos\delta_{f_i}\right) F_{x_{f,l}} \\ \qquad + \left(l_f \sin\delta_{f_i} + \frac{t}{2}\cos\delta_{f_i}\right) F_{x_{f,r}} \\ \qquad - \left(l_r \sin\delta_{r_i} + \frac{t}{2}\cos\delta_{r_i}\right) F_{x_{r,l}} \\ \qquad - \left(l_r \sin\delta_{r_i} - \frac{t}{2}\cos\delta_{r_i}\right) F_{x_{r,r}} \end{cases} \tag{2}$$

Where:

- s : Laplace operator,
- M : vehicle's mass,
- I_z : yaw inertia moment of the vehicle,
- t : vehicle's track,
- l_k : distance between the axle k and the vehicle's centre of gravity, where k designs the front or the rear,
- V_x : longitudinal speed,
- V_y : lateral speed,
- $\dot{\psi}$: vehicle's yaw rate,
- F_{y_k} : overall lateral force at the axle k,
- $F_{x_{k,j}}$: longitudinal force of each tire, where j designs left or right,
- δ_{k_i} : the initial steering angle before the application of the tire force.

Following equation (2), many research papers on CA in the automotive field consider a pure integrator relationship between the lateral force and the vehicle's yaw rate [5].

[1]Called extensively as the 4-Wheel Steering (4WS) system by car manufacturers.

However, the relationship between the lateral force and the yaw rate is actually nonlinear. Indeed, considering for example the new linear tire model with varying parameters [15], the lateral tire force can be defined as follows:

$$F_y = C^*_\alpha(\kappa, \mu, F_z)\,\alpha \tag{3}$$

Where:

- C^*_α : varying lateral stiffness that takes into account the longitudinal slip κ, the friction coefficient μ and the vertical load F_z in order to represent the combined slip phenomenon,
- α : the side-slip angle.

In addition, the side-slip angle is defined as follows:

$$\alpha_k = \delta_k - \frac{V_y \pm l_k\dot{\psi}}{V_x} \tag{4}$$

Therefore, the lateral tire force depends on the lateral speed and the yaw rate. A linear transfer function cannot be expressed between the lateral force in this form and the yaw rate. In the following, we will consider only the rear lateral force related to the 4WS system as our prototype is not autonomous yet. The front lateral force is influenced by the human driver and is used to generate the yaw rate reference as Fig. 1 shows. In order to synthesize a high level controller to generate a total yaw moment, we split the controllable term of the lateral force from the non-controllable one:

$$F_{y_r} = C^*_{\alpha_r}\delta_r - C^*_{\alpha_r}\frac{V_y - l_r\dot{\psi}}{V_x} = F_{y_{r_c}} + F_{y_{r_{nc}}} \tag{5}$$

$F_{y_{r_{nc}}}$ will be then integrated in the state-space matrix, and only $F_{y_{r_c}} = C^*_{\alpha_r}\delta_r$ is considered as an input. A high-level controller can be then synthesized to generate the total yaw moment, by considering the influence of the rear angle on the yaw moment as $M_{4WS} = -C^*_{\alpha_r} l_r \delta_r$. After few simplifications and combinations, we find as expected a second-order transfer function between the yaw moment and the yaw rate with a gain and two complex poles depending on the longitudinal speed [16]. Each input comes with a different zero. And finally, as the actuators comes from different suppliers with inner control algorithms in black boxes [14], we prefer to identify their dynamics experimentally.

III. EXPERIMENTAL IDENTIFICATION

The difficulty of experimental identification manifest in the few measurable signals as Fig. 2 illustrates. In Fig. 2, the accessible signals are in green, and the inaccessible ones are in red, where:

- $T_{b_{k,j_{req}}}$: requested brake torque at the $k-j$ wheel,
- $T_{b_{k,j_{eff}}}$: effective brake torque at the $k-j$ wheel,
- $\delta_{r_{req}}$: requested rear steering angle,
- $\delta_{r_{eff}}$: effective rear steering angle.

The effective brake torques applied are not measurable in our case. Therefore, a direct identification of the brake dynamics is not possible. In contrast, the effective rear steering angle is measurable. The idea is then to identify first the 4WS actuator dynamics and the vehicle dynamics by activating only the 4WS system. Then, when identifying the transfer function between $T_{b_{k,j_{req}}}$ and $\dot{\psi}$, we can isolate the already identified vehicle poles to deduce the brake dynamics.

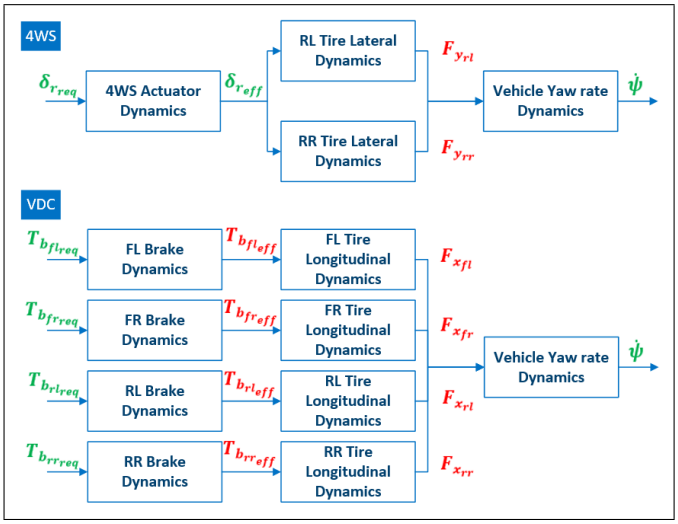


Fig. 2. The identification problem in an over-actuated vehicle.

A. The 4WS system identification

Vehicle dynamics depend on the vehicle's speed [17]. The experiments should be carried out at different speed values. We apply a step to the 4WS system, and we measure both the effective rear steering angle $\delta_{r_{eff}}$ and the yaw rate of the vehicle $\dot{\psi}$ to evaluate the influence of the 4WS actuator on the vehicle. We use afterwards the System Identification app of Matlab® to identify the dynamic models using input/output data. For a speed of $70km/h$ for example, we obtain the Fig. 3. We use different transfer function shapes to approach the

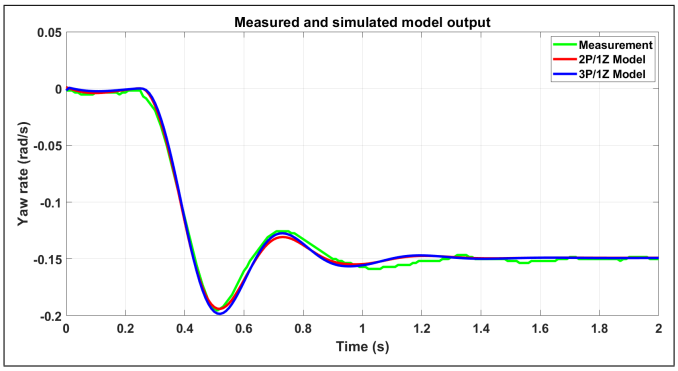


Fig. 3. Comparison of the measured response and the estimated ones.

vehicle response. After several experiments we can conclude that two poles and one zero suffice to represent the vehicle dynamics influenced by the rear steering. Fig. 3 shows that there is no need to add for example a third pole. The zero remains characteristic to the 4WS system. The real dynamics of the vehicle are the two identified poles, which falls right into our expectations. Using several experiments for different speed values, we can finally isolate the constant parameters from the vehicle's speed influence.

The same procedure is carried out for the 4WS actuator only. This time a small delay is needed to fit the real dynamics of the actuator as Fig. 4 shows. The overall system

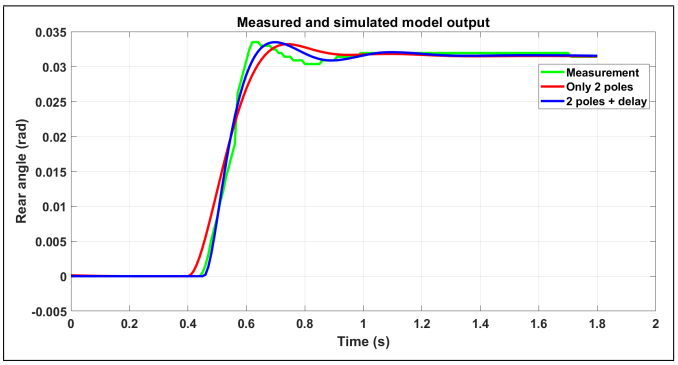


Fig. 4. Comparison of the measured response and the estimated ones for the 4WS actuator.

is composed then of two poles characterizing the vehicle dynamics, one zero, one delay and two poles characterizing the 4WS system dynamics.

B. The VDC system identification

We follow the same procedure for the VDC also. Here, we apply a torque of $400N.m$ at each left wheel, and we measure the yaw rate response of the vehicle. Fig. 5 shows that three

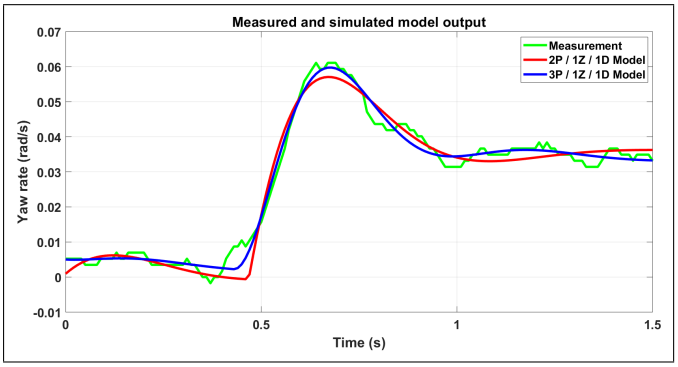


Fig. 5. Comparison of the measured response and the estimated ones for the VDC.

poles, one zero, and one delay are needed to represent both the vehicle and the VDC system.

By analyzing all the poles and zeros identified (see Fig. 6), we can see that the complex poles of both dynamic models are close enough. The complex poles characterize then the vehicle dynamics. The remaining pole, zero and pure delay characterize the VDC system. These identified models are then used to synthesize both the high-level controller and the MPCA.

IV. MODEL PREDICTIVE CONTROL ALLOCATION

First of all, let us define the control allocation problem [18]: find $\vec{u} \in \mathbb{R}^n$ such that

$$\mathbf{B}\vec{u} = \vec{v} \tag{6}$$

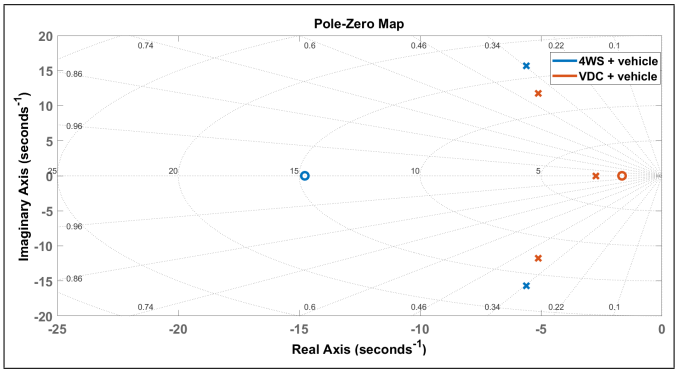


Fig. 6. Poles-zeros analysis.

subject to

$$\begin{cases} \vec{u}_{min} \leq \vec{u} \leq \vec{u}_{max} & (7) \\ \dot{\vec{u}} \leq \dot{\vec{u}}_{max} & (8) \end{cases}$$

where $\mathbf{B} \in \mathbb{R}^{m \times n}$ is a control effectiveness matrix, $\vec{u}_{min} \in \mathbb{R}^n$ and $\vec{u}_{max} \in \mathbb{R}^n$ are the lower and upper position limits, respectively, $\dot{\vec{u}} \in \mathbb{R}^n$ is the control rate, $\dot{\vec{u}}_{max} \in \mathbb{R}^n$ is the maximum control rate, $\vec{v} \in \mathbb{R}^m$ are the desired accelerations, n is the number of control effectors, and m is the number of axes to control with $n > m$.

In our case, $\vec{v} = M_{z_{tot}}$. However, the low control-level will differ from a CA approach to another as we will explain in the following paragraphs. In contrast, the same high-level controller can be adopted. As robustness is one of our main concerns, $\mathcal{H}_\infty$ synthesis is chosen as a high-level controller. It is an optimization method to minimize the $\mathcal{H}_\infty$ norm of the augmented plant containing weight functions. Three weight functions are considered here that penalize the error signal, control signal, and output signal, respectively. As the vehicle dynamics depends on the varying longitudinal speed, gain scheduling is then used to adapt the controller behavior to speed values. This, however, goes beyond the scope of this paper. The interested readers can refer to [19].

A. Classic Dynamic Control Allocation

Here, the control architecture exposed in Fig. 1 is carefully followed. The CA layer generate the virtual tire forces to be applied. Then, low-level controllers are added to take into account the zero of each input along with each actuator dynamics. Regarding this low-level layer, $\mathcal{H}_\infty$ synthesis can be adopted again. Nonetheless, the actuators' dynamics present pure input time-delays which add non-linearities in the process. $\mathcal{H}_\infty$ cannot be applied directly. Consequently, here we use second-order *Padé* approximation to approximate the time-delays.

As for the CA algorithm, online optimization techniques can be used. Various techniques have been compared in [8]. It has been shown that the Weighted Least Squares (WLS) formulation based on Active Set Algorithms (ASA) solves the problem rapidly with good precision and reach the optimum in a small finite number of iterations. The WLS solves the global problem in a one stage ASA by means

of different weights to determine the importance of each objective. This gives the following expression:

$$\vec{u}_{opt} = arg \min_{\vec{u}_{min} \leq \vec{u} \leq \vec{u}_{max}} \left\| \mathbf{W_u} \left(\vec{u} - \vec{u}_p \right) \right\|^2 + \gamma \left\| \mathbf{W_v} \left(\mathbf{B}\vec{u} - \vec{v} \right) \right\|^2 \quad (9)$$

Where:

- $\vec{u}_p$: preferred control vector[2],
- γ : weight that enables favouring an objective over another,
- $\mathbf{W_u}$: non-singular weighting matrix affecting control distribution among the actuators,
- $\mathbf{W_v}$: non-singular weighting matrix affecting the prioritization among the virtual control components when $\mathbf{B}\vec{u} = \vec{v}$ cannot be attained due to actuator constraints.

In our case, $\vec{u}$ is composed of tire forces, and $\mathbf{B}$ is composed of the geometrical coefficients shown in equation (2). The limits of the control vector are composed of the minimum of the actuator position and rate limits, and the friction ellipse concept [15].

B. Model Predictive Control Theory for Control Allocation

The previous method takes into account only the most effective effector in the distribution process. However, each system has a different dynamic. In severe maneuvers, the fastest system may constitute a better choice. For missile control for example, fast actuators can be used for good transient response, while slow actuators can be used for steady state flight and therefore improving the power efficiency [20]. Other interesting scenarios are when several systems should be activated because of the limits of each one. The less effective systems should be activated as soon as the most effective ones saturate. When the less effective systems present time-delays, as it is the case here, the command should be requested a few steps before the saturation of the most effective systems. To the best of our knowledge, this can only be ensured if a certain prediction of the integrated dynamics are provided.

The MPC is an online optimization-based control technique that aims to solve a finite-horizon optimization problem at each sampling time. An internal discretized dynamic model is used to predict the behaviour of the system, and the optimizer generates the required control inputs in order to satisfy the desired performances along a chosen prediction horizon. The MPC objective cost function is usually taken as follows [20]:

$$J(k) = \sum_{i=1}^{T} Q(i) \left(\hat{x}(k+i|k) - r(k+i|k) \right)^2 + \sum_{i=1}^{T} R(i) \left(\hat{u}(k+i|k) \right)^2 \quad (10)$$

Where k is the current time step, $\hat{x}$ is the estimated state, r is the reference trajectory, $\hat{u}$ is the optimal control sequence, and T is the prediction horizon length. The first term in $J(k)$ represents the reference tracking performance, and the second one represents the control effort mitigation. The weights $Q(i)$ and $R(i)$ enable favoring one objective over another. In addition, as far as an optimization problem is concerned, the MPC is capable of handling constraints at both the control input level and the state level. The principle of the MPC is illustrated in Fig. 7. The optimizer explores

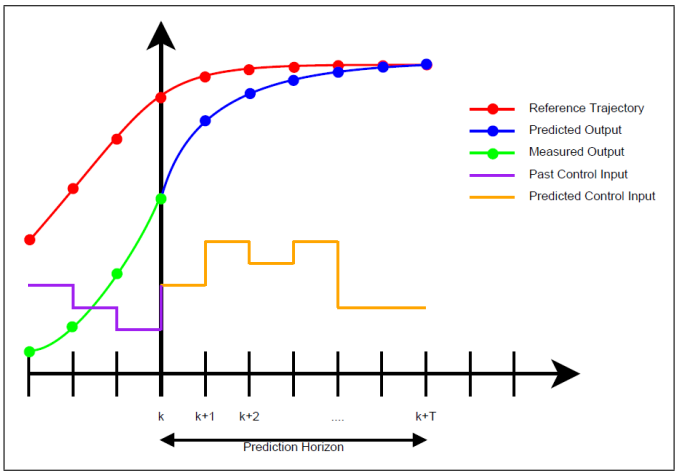


Fig. 7. An MPC general scheme (taken from [20]).

the state trajectories minimizing the cost over the prediction horizon, and only the first control step is applied to the plant. The procedure is repeated for every sampling time in case of changes in the varying parameters of the integrated model. This technique is also called the Receding Horizon Control (RHC) because the horizon keeps being pushed forward.

Regarding our CA problem, the same philosophy can be applied. Rather than imposing a specific state to follow a desired reference, we can modify the cost function to make the virtual forces track the required total yaw moment. What is more interesting in this approach, is that we can take into account the actuator dynamics with their delays in an explicit manner. The integrated model in that case is not the overall system, but only the actuators' models. Thanks to its predictive nature, this approach can pre-act on the actuators by taking into account the time-delay so the tire forces can be applied at the right moment. To do so, the actuators' models should be first discretized and combined with the virtual forces expressions. Here, we use Tustin method:

$$\begin{cases} F_{x_{i,j}}(k) = a_b F_{x_{i,j}}(k-1) \\ \qquad + b_{b_1}(V_x) T_{b_{i,j_{req}}} \left(k - \frac{\tau_{vdc}}{T_s} - 1 \right) \\ \qquad + b_{b_2}(V_x) T_{b_{i,j_{req}}} \left(k - \frac{\tau_{vdc}}{T_s} - 2 \right) \end{cases} \quad (11)$$

$$\begin{cases} F_{y_r}(k) = a_{r_1} F_{y_r}(k-1) + a_{r_2} F_{y_r}(k-2) \\ \qquad + b_{r_1}(V_x) \delta_{r_{req}} \left(k - \frac{\tau_{4ws}}{T_s} - 2 \right) \\ \qquad + b_{r_2}(V_x) \delta_{r_{req}} \left(k - \frac{\tau_{4ws}}{T_s} - 3 \right) \end{cases} \quad (12)$$

Where the parameters a_i and b_i are those obtained by

[2]Usually $\vec{0}$ due to energy consumption

discretization. Note that the parameters b_i depend on the vehicle's speed. This is due, as we have mentioned, to the dependency of the zeros on the speed. τ_{vdc} and τ_{4ws} are the input time-delay of the VDC and 4WS systems respectively. And finally, T_s is the sampling time.

Let us define the following vectors:

$$\begin{cases} \vec{u} = \begin{bmatrix} F_{x_{f,l}} & \dots & F_{x_{r,r}} & F_{y_r} \end{bmatrix}^t & (13) \\ \vec{\delta} = \begin{bmatrix} T_{b_{f,l_{req}}} & \dots & T_{b_{r,r_{req}}} & \delta_{r_{req}} \end{bmatrix}^t & (14) \end{cases}$$

Where the superscript t means the transpose. By adapting the MPC problem to our CA problem, we can define the MPCA problem as: find $\vec{\delta} \in \mathbb{R}^n$ such that

$$\mathbf{B}\vec{u} = \vec{v} \tag{15}$$

subject to

$$\begin{cases} \dot{\vec{u}} = A_d\vec{u} + B_d\vec{\delta} & (16) \\ \vec{u}_{min} \leq \vec{u} \leq \vec{u}_{max} & (17) \\ \dot{\vec{u}} \leq \dot{\vec{u}}_{max} & (18) \\ \vec{\delta}_{min} \leq \vec{\delta} \leq \vec{\delta}_{max} & (19) \\ \dot{\vec{\delta}} \leq \dot{\vec{\delta}}_{max} & (20) \end{cases}$$

Where A_d and B_d are filled with the discretized parameters a_i and b_i respectively according to equations (11) and (12). The limits imposed to $\vec{u}$ are due to the friction ellipse [15], while the limits imposed to $\vec{\delta}$ are due to the actuators' saturations. Since the algorithm generates directly the actuator commands, the mild-level layer and the low-level layer in Fig. 1 are now merged into one layer as Fig. 8 shows. Following the MPC framework, the cost function

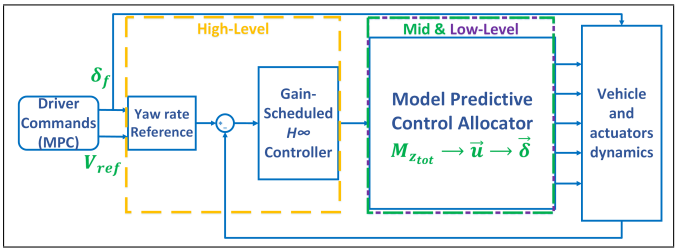


Fig. 8. Multi-layered control architecture in case of MPCA.

to be minimized here is therefore:

$$J_{mpca}(k) = \sum_{i=1}^{T} \left(\sum_{j=1}^{5} B_j u_j(k+i) - v(k+i) \right)^2 + \gamma \sum_{i=1}^{T} \sum_{j=1}^{5} W_u(j) \left(\delta_j(k+i)\right)^2 \tag{21}$$

The major motivation that drove us to use MPC theory for CA problems is the presence of input time-delays. In this context, the prediction horizon T should be at least superior to the bigger time-delay. In our case, the VDC system has a time-delay of almost 180ms while the 4WS system has a time-delay of almost 50ms. Therefore, we choose $T = 200ms$. The idea is then to compute the evolution of states through the prediction horizon, and choose as an output the commands that should be applied at the time step k by taking into account the time-delays. This is especially relevant when the 4WS system, as the most effective system [1], saturates, and the VDC system has to be activated instantly afterwards. Following the same representation as in Fig. 7, an illustration of our case is presented in Fig. 9. We can see that thanks to

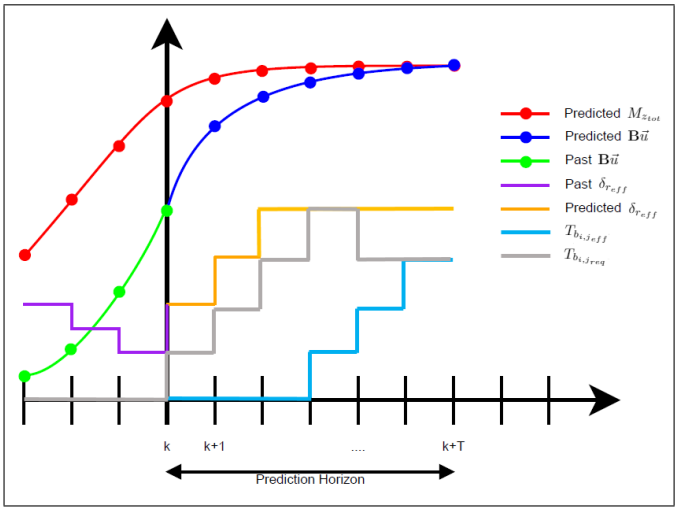


Fig. 9. The MPCA scheme (adapted from [20]).

the prediction of the saturation of the effective rear steering angle, the algorithm can request the braking torque few steps before the saturation taking into account the delay and raising time of the VDC. In this way, as soon as the 4WS is saturated, effective braking torques can be applied and complete the maximum effort generated by the 4WS system.

Finally, one major difference with respect to other researches in MPCA [11], [12], [20], is the solver used to solve the problem. In most of these researches, the computation effort constitute one of the major drawbacks. As shown in [8], the ASA solver coupled with the WLS formulation gives fast and reliable results in a limited number of iterations. Therefore, ASA is again used here to solve the MPCA problem. The only additional effort required from the control designer is to transform the problem in a WLS formulation.

V. SIMULATION RESULTS

As a first step, the algorithm should be first validated by simulation. However, in order to present realistic simulation results, we test the control architecture using the experimentally identified models via Matlab R2017b® in a CORE i7 7^{th} generation machine. One can think of using co-simulation of Matlab® and a high-fidelity software as LMS Imagine.Lab AMESim® [8], [12], however, even these high-fidelity softwares do not encompass the latest advanced chassis systems with their real dynamics.

In order to evaluate the relevance of the MPCA, we test both the classic dynamic CA and the MPCA when the 4WS saturates. This could happen when avoiding a pop-up obstacle and turning the front steering wheel really fast. We represent simply this situation as a step command applied

to the yaw rate target. The results for both methods are illustrated in Fig. 10.

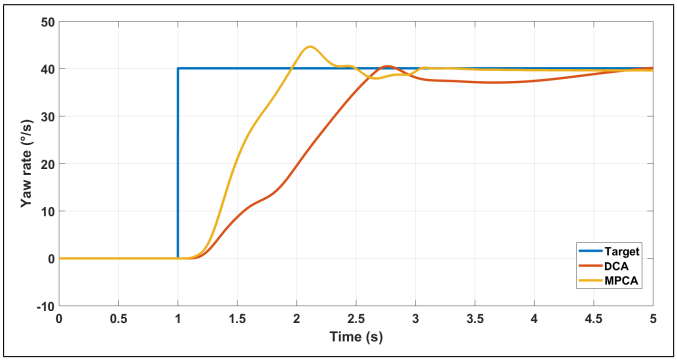


Fig. 10. Comparison of the classic dynamic CA and the MPCA.

As Fig. 10 shows, the response of the vehicle using the MPCA is faster than the classic dynamic CA. To understand the reason behind this, we analyse the effective commands in each approach. The effective rear angle and the effective front-left brake torque[3] in case of the classic CA is depicted in Fig. 11. Because the control allocator acts only after the saturation of the 4WS system, the VDC is activated only after its characteristic input time-delay of 180ms. This explains also the deflection of the yaw rate signal in case of the classic dynamic CA in Fig. 10.

In contrast, as Fig. 12 shows, this is not the case when using the MPCA algorithm. Thanks to the prediction ability of the MPCA and the precise description of the actuators' dynamics, the request of brake torques is generated before the 4WS saturation. Not only the brake torques are activated as soon as the 4WS is saturated, but they are activated few steps before in order to get the right amount of the brake torques after the saturation.

VI. INDUSTRIAL LIMITS

Our motivation is to make commercial passenger cars safer, more comfortable, and increase their performance. However, all these advantages come with a price, which is an increased complexity. In general, this induces a higher computation effort demand in case of the existence of a solution. For the MPC theory, an additional challenge is added, which is robustness.

A. Computation Effort

Classic dynamic CA algorithms run an optimization solver online. This is already a challenging task when it comes to the automotive control where the dynamics can change in milliseconds. Regarding the MPCA, the same procedure should be redone through all the prediction horizon. If the prediction horizon is 200ms, and if we take a sampling time of 10ms, 20 CA problems should be solved at each step. Needless to say that faster and more expensive Electronic Control Units (ECUs) are required. The increase in cost should be justified by the additional features that the MPCA provides. If fault-tolerance is required between the embedded systems, and if a redundant system is more expensive than a faster ECU, then the choice of the MPCA is cost-effective.

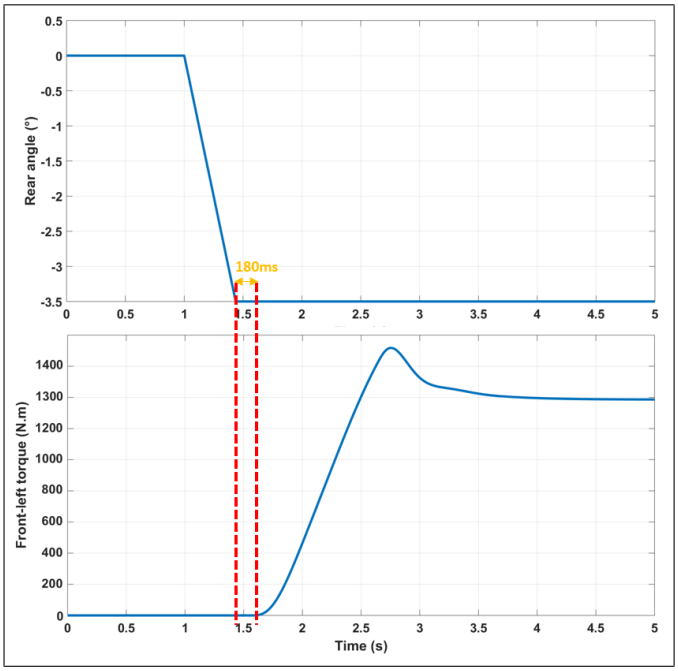


Fig. 11. The commands in case of classic dynamic CA.

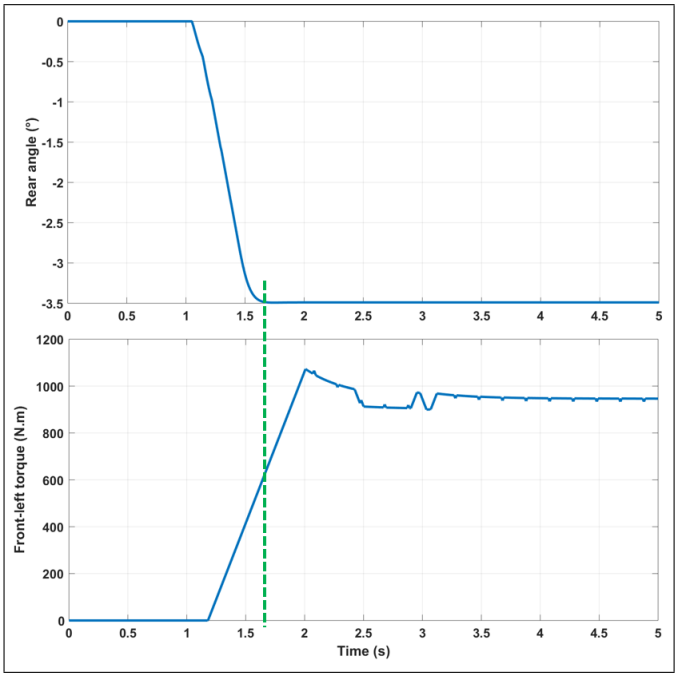


Fig. 12. The commands in case of MPCA.

Another option is to well tune the MPCA algorithm. The designer can reduce the prediction horizon. However, in the presence of time-delays, these latter impose a minimum value

[3]In this scenario, only the left brakes are activated to generate a positive yaw rate. The rear brakes are not illustrated because they have the same shape as the front brakes with just a lower amplitude.

to the prediction horizon. The other alternative is reducing the MPC sampling time. The drawback is that the command becomes more aggressive. Last but not least, is to solve the problem offline when it is possible. This is called the explicit MPC [21]. However, for multi-objectives problems, an analytic solution could be hard to determine. This falls right in our problem as we are aiming for safety, comfort and performance when it is possible. A compromise should be found between the performance of the algorithm and the cost of the ECU.

B. Robustness

The second drawback is the robustness. The performance of the MPCA is closely related to the implemented model. The more precise this model is, the better the control performances are, and vice-versa. This is the reason that drove us to choose the experimentally identified models to design the MPCA, and a high-level $\mathcal{H}_\infty$ controller. Nevertheless, important effort have been deployed in the field of Robust MPC. To introduce robustness, a family of systems is considered by adding an unmeasured noise as an input to the system. A survey on Robust MPC is provided in [22]. The same techniques can be expected for the MPCA.

Adding robustness in this way may reduce the performances of the controller. Another solution could be the online identification of the plant model in the same style as the Model Reference Adaptive Control (MRAC) [23]. The problem in our case is that we need the identification of the actuators models. In this case, we need both the request signal and the effective signal at the input and output of each actuator repectively. Regarding the VDC system, we do not measure the effective torque brakes due to the cost of their sensors. An additional challenge of estimating the effective brake torque is expected. With online identification, online optimization and prediction, the complexity might become unmanageable. Both Robust MPC and MRAC theories are expected to be investigated in our upcoming works.

VII. CONCLUSIONS

In this paper, a Model Predictive Control Allocation algorithm has been developed for systems with different dynamics and time-delays. This algorithm has been compared to the classic dynamic control allocation algorithm. The MPCA has proven its relevance especially in the presence of saturation of one of the chassis systems. The prediction of this saturation leads to the activation of the secondary systems at the right moment. This MPCA method requires however higher computation effort and lacks of robustness. Due to the attractive features that this method offers, our future works are focused on the reduction of computation effort through the optimization of the solver, as well as the improvement of the control logic robustness.

REFERENCES

[1] M. Kissai, B. Monsuez, D. Martinez, X. Mouton and A. Tapus, A Comprehensive Comparison of Chassis Systems Coordination Approaches, 2018 18th International Conference on Control, Automation and Systems (ICCAS), Daegwallyeong, 2018, pp. 351-356.

[2] Marc Bodson, Evaluation of Optimization Methods for Control Allocation, Journal of Guidance, Control, and Dynamics, Vol. 25, No. 4 (2002), pp. 703-711.

[3] M. W. Oppenheimer, D. B. Doman, and M. A. Bolender, Control Allocation for Over-actuated Systems, 2006 14th Mediterranean Conference on Control and Automation, Ancona, 2006, pp. 1-6.

[4] L. Bai, Z. Gao, M. Qian and B. Han, Fault tolerant scheme design for the flight control systems using adaptive control allocation technique, 2018 Chinese Control And Decision Conference (CCDC), Shenyang, 2018, pp. 2247-2251.

[5] A. Soltani, Low Cost Integration of Electric Power-Assisted Steering (EPAS) with Enhanced Stability Program (ESP), PhD thesis, Cranfield University, 2014.

[6] G. De Filippis, B. Lenzo, A. Sorniotti, P. Gruber and W. De Nijs, Energy-Efficient Torque-Vectoring Control of Electric Vehicles With Multiple Drivetrains, in IEEE Transactions on Vehicular Technology, vol. 67, no. 6, pp. 4702-4715, June 2018.

[7] E.H. van den Berg, Optimal Control Allocation on Over-Actuated Vehicles, Master thesis, Delft Center for Systems and Control (DCSC), August 2016.

[8] M. Kissai, X. Mouton, B. Monsuez, D. Martinez and A. Tapus, Complementary Chassis Systems for Ground Vehicles Safety, 2018 IEEE Conference on Control Technology and Applications (CCTA), Copenhagen, 2018, pp. 179-186.

[9] W. M. Haddad, V. Kapila and C. T. Abdallah, Stabilization of linear and nonlinear systems with time delay, Proceedings of the 1997 American Control Conference (Cat. No.97CH36041), Albuquerque, NM, USA, 1997, pp. 3220-3224 vol.5.

[10] D. Yue, Delayed feedback control of uncertain systems with time-varying input delay, Automatica, vol. 41, issue 2, 2005, pp. 233-240.

[11] C. Satzger et al., A model predictive control allocation approach to hybrid braking of electric vehicles, IEEE Intelligent Vehicles Symposium Proceedings, Dearborn, MI, 2014, pp. 286-292.

[12] H. Zhao, B. Ren, H. Chen and W. Deng, Model predictive control allocation for stability improvement of four-wheel drive electric vehicles in critical driving condition, in IET Control Theory & Applications, vol. 9, no. 18, pp. 2688-2696, 10 12 2015.

[13] R. Anderson and D. M. Bevly, Using GPS with a model-based estimator to estimate critical vehicle states, Vehicle System Dynamics, Taylor & Francis, vol. 48, no. 12, 2010, pp. 1413-1438.

[14] M. Kissai, B. Monsuez and A. Tapus, Review of integrated vehicle dynamics control architectures, 2017 European Conference on Mobile Robots (ECMR), Paris, 2017, pp. 1-8.

[15] M. Kissai, B. Monsuez, A. Tapus and D. Martinez, A new linear tire model with varying parameters, 2017 2nd IEEE International Conference on Intelligent Transportation Engineering (ICITE), Singapore, 2017, pp. 108-115.

[16] J. R. Ellis, Vehicle Dynamics, F.I.MECH.E. Advanced School of Automobile Engineering, Cranfield, 1969.

[17] E. Ono, K. Takanami, N. Iwama, Y. Hayashi, Y. Hirano, Y. Satoh, Vehicle integrated control for steering and traction systems by μ-synthesis, Automatica, vol. 30, issue 11, 1994, pp. 1639-1647.

[18] T. A. Johansen and T. I. Fossen, Control Allocation - A survey, in Automatica, Vol. 49, Issue 5, May 2013, pp. 1087-1103.

[19] M. Kissai, B. Monsuez, A. Tapus, X. Mouton, and D. Martinez, Gain-Scheduled $\mathcal{H}_\infty$ for Vehicle High-Level Motion Control, in Proceedings of the 6th International Conference on Control, Mechatronics and Automation (ICCMA 2018), ACM, New York, NY, USA, 97-104.

[20] M.B. Hanger, Model Predictive Control Allocation, Master thesis, Institutt for teknisk kybernetikk, Norwegian University of Science and Technology, 2011.

[21] A. Alessio, A. Bemporad, A Survey on Explicit Model Predictive Control, Nonlinear Model Predictive Control: Towards New Challenging Applications, Springer Berlin Heidelberg, 2009.

[22] A. Bemporad and M. Morari, Robust model predictive control: A survey, Robustness in identification and control, Springer, 1999.

[23] A. Alan, Y. Yildiz and U. Poyraz, High-Performance Adaptive Pressure Control in the Presence of Time Delays: Pressure Control for Use in Variable-Thrust Rocket Development, in IEEE Control Systems Magazine, vol. 38, no. 5, pp. 26-52, Oct. 2018.