2014 IEEE Intelligent Vehicles Symposium (IV)
June 8-11, 2014. Dearborn, Michigan, USA

A Model Predictive Control Allocation Approach to Hybrid Braking of Electric Vehicles

Clemens Satzger, Ricardo de Castro, Tilman Bünte
German Aerospace Center (DLR), Institute of System Dynamics and Control
Muenchnerstr. 20, 82234 Wessling, Germany

Abstract **With the recent emergence of electric drivetrains, a faster and energy efficient braking actuator – the electric motor – has become available to complement the operation of the traditional friction brakes. The decision on how to split the braking torque among the friction brake and the electric motor is one of the main issues of such hybrid braking systems. With this challenge in mind, a new model predictive control allocation (MPCA) approach for hybrid braking is proposed. In comparison to state of the art torque blending solutions (daisy chain and dynamic control allocation) the MPCA offers faster transient response, without compromising the energy recuperation efficiency of the actuators. In addition, we also develop a linear wheel slip controller to regulate the braking force during emergency braking maneuvers. The tuning of this wheel slip controller is carried out using robust pole placement techniques, which ensures good operation in spite of uncertainties in the tire-road friction coefficient and the vertical load. Simulation results demonstrate the effectiveness of the proposed method.**

Keywords: wheel slip control; control allocation; model predictive control allocation; brake blending; recuperation

I. INTRODUCTION

The evolving class of electric vehicles (EV) with near-wheel motors and brake-by-wire provides new possibilities of motion control, such as torque vectoring and hybrid braking.

Our robotic vehicle ROMO [1], with four in-wheel motors (IWM), represents an example of such powertrains. In these powertrains, the regenerative capabilities of the traction motors can be used to support the brake-by-wire friction brake (FB) system during braking. In this situation the motor is used as a generator and recharges the batteries or capacities of the EV. Therefore, from an energy-consumption standpoint, it is preferable to maximize the usage of the motor torque during braking. Moreover, due to the fast and precise response of the traction motor, this electric actuator can also be used to improve the bandwidth of the wheel torque and wheel slip control [2].

From a control point of view, the braking system of an EV can also be regarded as an over-actuated system. This is because the EV's braking force is realized by a redundant set of actuators: traction motor(s) and friction brakes. Static control allocation (CA) [3] represents a well-known technique to cope with such redundant actuation system. This formulation allows dealing with actuator failures and its saturation. However, no actuator dynamics are explicitly considered in this formulation and, therefore, the application of CA is unable to extract the maximum benefit from hybrid braking systems.

To attenuate these limitations, dynamic control allocation (DCA) has been proposed [4]. The main idea is to add a dynamic penalty in the cost function - to be optimized - and adapt the inequality constraints when the actuator rate limits occur. Reference [2] applies this approach to the *torque blending* problem, i.e. the split of braking torque among the friction brake and the electric motor. Albeit being capable of improving the transient response of the hybrid brake system, the DCA algorithm relies on an *ad-hoc* strategy to implicitly cope with the actuator dynamic model.

The explicit incorporation of actuator's dynamics models becomes possible using the third approach examined here: model predictive control allocation (MPCA). This approach combines Model Predictive Control (MPC) with CA. Thus, it is an optimization-based CA approach that can handle actuator dynamics (formulated as equality constraints) and range and rate limitations (posed as inequality constraints). The explicit consideration of the actuator's dynamic model allows inverse pre-allocation of the control setpoint, which improves the individual actuator dynamics. Motion control of launch and reentry vehicles, marine vessels and thermal management problems represent recent examples where the MPCA has been successfully applied [5] [6] [5] [7] [8] [9]. In this context, the first contribution of this paper consists in the application of the MPCA technique to the torque blending problem.

An additional goal of this paper consists in the design of a practical, but robust, linear wheel-slip controller. Reference [10] reports a static linear well-slip controller to prevent excessive wheel slip. However, this controller cannot take into account the large variations of the tire road uncertainties. To mitigate this issue, variable-gain linear controllers, based on velocity-based variable gain Proportional+Integral strategies, have been reported in the literature [11], [12]. The idea of the velocity-dependent controller gains will be extended in this paper using a new formulation, based on robust pole placement techniques.

II. MODELLING AND CONTROL FRAMEWORK

A. Overview of the Control System

In this work, we will consider an electric vehicle (EV) equipped with four IWM and four FB, similar to the ones employed in the ROMO vehicle [1]. It will also be assumed that the EV is endowed with a vehicle dynamics controller (VDC), responsible for translating the motion demands specified by the driver (e.g., velocities and/or forces) into wheel steering and wheel torque set points, as illustrated in Figure 1 and described in [13], [14]. Based on the wheel torque set points, it becomes necessary to find an appropriate splitting strategy between electric (T_e) and friction brake torques (T_f), i.e. the *torque blending*. Towards that goal, we will follow here a methodology similar to the one resented in [2]. More specifically, each wheel of the vehicle has its

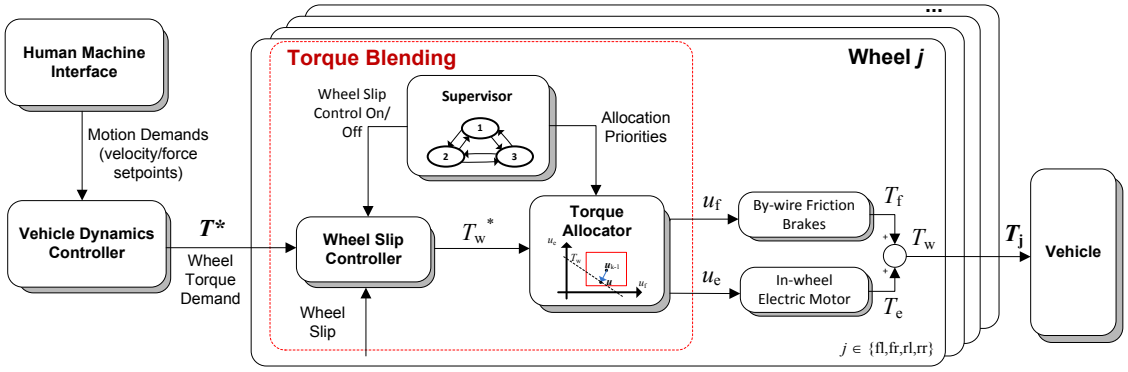

Figure 1: Control architecture employed in this work

associated torque blending block, composed of three sub-components (see Figure 1):
i) a wheel slip controller; ii) a torque allocator; and iii) a supervisor. The first block, the wheel slip controller, manipulates the wheel torque set point in case of emergency braking in order to prevent excessive - and potentially dangerous - tire-slip. The second block, the torque allocator, distributes the wheel slip controller torque demands T_w^* among the two braking actuators. While doing so, the aim is to maximize the energy recuperation during normal braking (i.e. when $T_w^* = T^*$), and the promotion of a good transient response and slip control performance in wheel slip controller operation. Finally, the supervisor manages the total torque blending process by activating/deactivating the wheel slip controller and adapting the allocation parameterization. In the remainder of this section we will describe practical control-oriented models for the vehicle longitudinal dynamics and actuators, which will be used later on for the design of the torque blending controller.

B. Control-Oriented Modeling

As a starting point for the modeling, let us consider the single corner model, widely used in the literature, e.g. [15].

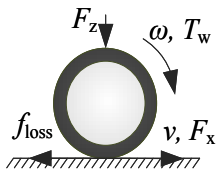

Figure 2: Single corner

$$J\dot{\omega} = -r\mu(\lambda)F_z + T_w \quad (1)$$

$$m\dot{v} = \mu(\lambda)F_z - f_{loss} \quad (2)$$

$$\lambda = \frac{\omega r - v}{v} = h(\omega, v) \quad (3)$$

$$\mu(\lambda) = \frac{F_x}{F_z} \quad (4)$$

The first equation (1) accounts for the wheel rotational dynamics as defined in Figure 2, where J is the wheel-inertia, ω is the wheel rotational speed, r is the wheel radius, $\mu(\lambda)$ the slip dependent friction coefficient defined in equation (4) and T_w is the wheel drive or braking torque. The second equation (2) is related with the translational speed v of the car, where m is the mass of the quarter vehicle, f_{loss} contains aero, dragging and other losses in the car. The tire slip λ according to (4) will be the controlled variable for the wheel slip controller. To take into account the actuation dynamics, the above representation is augmented with the following control-oriented models of the electric motor and the friction brakes.

$$T_w = T_e + T_f \quad (5)$$

$$T_i^* = k_i u_i (t - \tau_{d_i}) \quad (6)$$

$$\tau_i \dot{T}_i(t) = T_i^*(t) - T_i(t), i \in \{e, f\} \quad (7)$$

$$\underline{T}_i \le T_i \le \overline{T}_i, \quad |\dot{T}_i| \le \overline{\dot{T}}_i, i \in \{e, f\} \quad (8)$$

Here, u_i is the control set point for the motor (index $i = e$) and the brake ($i = f$) that is transmitted to the actuation layer. The parameter τ_i represents the time-constant of the actuator response, k_i its gain and τ_{d_i} a dead time, which may appear due to communication delays and other factors. Finally, $\underline{T}_i$ and $\overline{T}_i$ are the actual minimum and maximum braking torques that each actuator can provide (cf. [2]) and $\overline{\dot{T}}_i$ the maximum rate limits. Equation (7) describes the linear actuator behavior and equation (8) the actuator limits. In the case the actuator is in its limits, equation (7) loses its validity.
The parameters employed in the single corner model are defined in Table I. Inspecting these values, there are three noteworthy facts. First, $\tau_e \ll \tau_f$. Consequently, the transients of the friction brake can be improved by the electric motor. Second, the friction brakes can provide higher braking torques than the electric motors, i.e. $\overline{T}_f > \overline{T}_e$. This factor makes the deployment of the friction brakes unavoidable during strong braking maneuvers.
Using the Modelica optimization library [16], we managed to parameterize the control oriented models of (7) and (8) with a R^2-fit of 99% on our brake testbench as demonstrated with Figure 3.

III. Wheel Slip Controller

In this work, a wheel slip controller was designed following

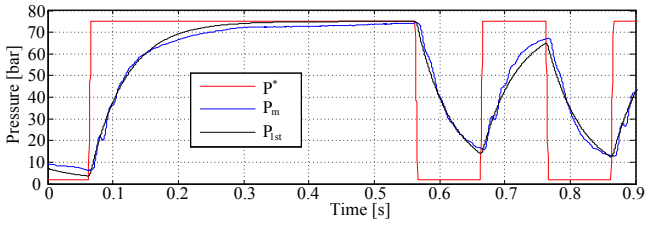

Figure 3: Comparison of the pressure responses of the control-oriented model from (7) denoted as P_{1st} with the real by-wire friction brake (P_m: measured pressure). P^* is setpoint.

a wheel-slip regulation setting based on the single corner model (1)-(4). The aim is to manipulate the wheel braking torque during emergency braking such that the tire slip follows a setpoint λ^*. A good choice for this setpoint is the supposed value of λ where the tire-road friction coefficient $\mu(\lambda)$ takes on its maximum (e.g., see [17] for additional details). By differentiating (3), inserting (1), (2), and assuming here $f_{\text{loss}} = 0$, yields

$$\dot{\lambda} = \frac{\partial h(\omega, v)}{\partial t} = \gamma \cdot (\Gamma(\lambda) + T_{\text{w}}) \tag{9}$$

which uses the abbreviations

$$\Gamma(\lambda) = -\left(1 + (1+\lambda)\frac{J}{mr^2}\right) \cdot r\mu(\lambda)F_{\text{z}} \; ; \; \gamma = \frac{r}{Jv} \tag{10}$$

according to [2], [15].

A. Linearization of the Wheel Slip

To address the abovementioned control problem, we need a linear model of the wheel slip. Therefore, we linearize (11) around the equilibrium point. As long as the term γ does not become zero (i.e. assuming non-zero speed), the equilibrium point of wheel slip (i.e. $\dot{\lambda} = 0$), characterized by the pair $(T_{\text{eq}}, \lambda_{\text{eq}})$, must fulfill the constraint $T_{\text{w,eq}} = -\Gamma(\lambda_{\text{eq}})$ in (9). Now, let us define the input and output deviations with respect to this equilibrium point,

$$\delta\lambda = \lambda - \lambda_{\text{eq}}, \quad w = T_{\text{w}} - T_{\text{w,eq}} \tag{11}$$

By linearizing the model (9) around this equilibrium point, we obtain the following approximation:

$$\dot{\delta\lambda} = \frac{r}{Jv} \cdot \left(\frac{\partial \Gamma}{\partial \lambda}\delta\lambda + w\right) = \gamma \cdot (\theta\delta\lambda + w) \tag{12}$$

$$\text{with} \quad \theta = \left.\frac{\partial \Gamma}{\partial \lambda}\right|_{\lambda=\lambda_{\text{eq}}} \tag{13}$$

Subsequently, we will regard the vehicle speed v as a slowly-varying known parameter; this approach is justified by the fact that the tire slip dynamics normally are much faster than the vehicle velocity dynamics [15]. Furthermore, we will consider θ as an uncertain parameter that lies in the range $[\underline{\theta}, \overline{\theta}] \subset \mathbb{R}$. Effective bounds $\underline{\theta}, \overline{\theta}$ can be determined by variation of all uncertain parameters that affect the partial derivative of $\Gamma(\lambda)$ according to (13).

Table I: PARAMETERS OF THE BRAKING MODEL

Variable	Symbol	Value
Wheel inertia	J	0.8 kg/m ^{2}
Wheel radius	r	0.274 m
Quarter car mass	m	262.5 kg
IWM time-constant	τ_e	300 μs
IWM gain	k_{e}	1 Nm/Nm
IWM delay	$\tau_{d_{\text{e}}}$	8 ms
Max. IWM torque	$\overline{T}_{\text{e}}$	160 Nm
Min. IWM torque	$\underline{T}_{\text{e}}$	−160 Nm
IWM rate limit	$\overline{\dot{T}}_{\text{e}}$	200 kNm/s
FB time constant	τ_{f}	54 ms
FB gain	k_{f}	−4.45 Nm/bar
FB delay	$\tau_{d_{\text{f}}}$	8 ms
Max. FB torque	$\overline{T}_{\text{f}}$	445 Nm
Min. FB torque	$\underline{T}_{\text{f}}$	0 Nm
FB rate limit	$\overline{\dot{T}}_{\text{f}}$	10,5 kNm/s

The transfer function of the linearized wheel-slip dynamics is extracted from (12) using the Laplace transformation.

Despite its simplicity, it is important to stress that the stabilization of this plant is not trivial. This is mainly due to the high level of parameter uncertainty that affects the plant's model, and which modifies its transient and stability characteristics.

$$\frac{\Delta\Lambda(s)}{W(s)} = \frac{\frac{1}{\overline{\theta}}}{\frac{1}{\gamma\overline{\theta}}s - 1} \tag{14}$$

B. Controller Design

The goal of the wheel-slip controller is to manipulate the torque difference w such that the slip difference $\delta\lambda$ reaches the difference setpoint $\delta\lambda^* = \lambda^* - \lambda_{\text{eq}}$, with some pre-specified dynamics, in spite of the model's parameter uncertainty. Notice that, in practice, we normally have a difference setpoint equal to zero, $\Delta\Lambda^* = 0$, i.e. the slip setpoint is the same as the equilibrium point selected for the linearization; nonetheless, non-zero difference setpoints may also be applied. In order to fulfill the control goal we will explore here a gain-scheduling proportional+integral (PI) control law:

$$\frac{W(s)}{\Delta\Lambda^*(s) - \Delta\Lambda(s)} = K_{\text{P}}(v) + \frac{K_{\text{I}}(v)}{s} \tag{15}$$

where K_{P} and K_{I} are the proportional and integral gains, which may vary slowly with the vehicle speed. To design (the scheduling of) the controller's feedback gains we can start by combining (15) together with the linearized plant (14). This gives the characteristic polynomial of this closed loop system.

$$s^2 + s\underbrace{\left(K_{\text{P}}(v)\frac{r}{Jv} - \frac{r}{Jv}\theta\right)}_{2\xi(v)\omega_{\text{n}}(v)} + \underbrace{K_{\text{I}}(v)\frac{r}{Jv}}_{\omega_{\text{n}}^2(v)} = 0 \tag{16}$$

In other words, this is a second-order system with natural frequency $\omega_{\text{n}}(v)$ and damping $\xi(v)$. Our design assumes robust pole placement where the natural frequency of the closed loop matches a desired value and the damping should be higher than a minimum value for all $\theta \in [\underline{\theta}, \overline{\theta}] \subset \mathbb{R}$:

$$\omega_{\text{n}}(v) = \omega_{\text{n}}^*(v) \tag{17}$$

$$\xi(v) \geq \xi^*(v) \tag{18}$$

For corresponding eigenvalue locations, a set of circular arcs, each for a distinct speed, is displayed in the complex plane (see Figure 4).

Solving (16), using (17), (18) for the controller gains, yields

$$K_{\text{I}}(v) = \frac{Jv\omega_{\text{n}}^*(v)^2}{r} \tag{19}$$

$$K_{\text{P}}(v) \geq \frac{2Jv\xi^*(v)\omega_{\text{n}}^*(v)}{r} + \overline{\theta} \tag{20}$$

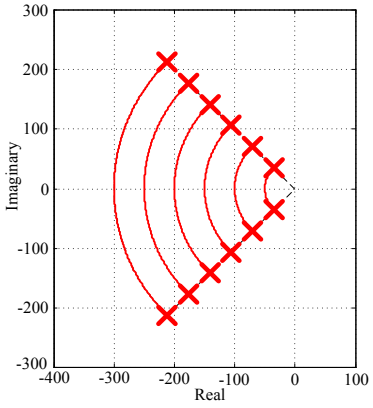

Figure 4: Specified parameter space with poles at 1, 1.2, 1.5, 2, 3 and 6m/s

Concluding from (20) only the upper limit $\overline{\theta}$ of θ is relevant for stability. Considering the Burckhardt tire model, $\overline{\theta} = 162$. Henceforth, we adopt a desired natural frequency that depends on the inverse of v and a minimum damping which is independent of the speed, i.e.:

$$\omega_{\mathrm{n}}^{*}(v) = \frac{\omega_{\mathrm{r}}}{v}, \qquad \xi^{*}(v) = \xi_{\mathrm{r}} \tag{21}$$

This way, the change of the closed loop bandwidth $\omega_{\mathrm{n}}(v)$ with speed follows the natural hyperbolic dependency of the open loop plant. Further, this approach also limits the control effort over the whole operating domain. Robust performance in terms of closed loop stability can be ensured by a sufficiently high minimum damping ξ_r. The values of ω_{r} and ξ_{r} represent tuning parameters that the designer can specify.

IV. Torque Blending Strategies

This section will discuss the allocation strategies for splitting (i.e., blending) the braking torque among the two braking actuators. In general this distribution is not unique since a wheel braking torque demand T_{w}^{*} can be realized by many combinations of $T_{\mathrm{f}} + T_{\mathrm{e}}$. Therefore, the challenge of allocation is to exploit the present redundancy. In the context of hybrid braking in EVs, this redundancy is often only utilized to reduce the energy consumption of the EV, i.e. to maximize the energy recuperation. Here, we are also striving for dynamic performance improvement by additionally regarding the different bandwidths and constraints (8) of each braking actuator. In this section, we will discuss three strategies to handle this allocation problem: i) Daisy Chain (DC), ii) Dynamic Control Allocation (DCA), and iii) Model Predictive Control Allocation (MPCA). The first two approaches represent state-of-the-art solutions, while MPCA is a new strategy that is proposed in the present article.

A. Daisy Chain (DC)

Roughly speaking, the recipe for the daisy chain method can be described as: employ the IWM, i.e. the most energy-efficient actuator, as long as possible, and only add the use of friction brakes (the less energy efficient device) when the electric torque reaches its saturation. Neglecting the actuation dynamics and rate limits, this approach can be mathematically described as:

$$T_{\mathrm{e}}^{*} = \mathrm{sat}_{[\underline{T_{\mathrm{e}}},\overline{T_{\mathrm{e}}}]}(T_{\mathrm{w}}^{*}) \tag{22}$$

$$T_{\mathrm{f}}^{*} = \mathrm{sat}_{[\underline{T_{\mathrm{f}}},\overline{T_{\mathrm{f}}}]}(T_{\mathrm{w}}^{*} - T_{\mathrm{e}}^{*}) \tag{23}$$

where $\mathrm{sat}_{[\underline{X},\overline{X}]}(X)\colon \mathbb{R} \to [\underline{X},\overline{X}]$ represents a saturation function block that constrains the passing signal to the bounds $[\underline{X},\overline{X}]$.

This static control allocation approach maximizes the energy recuperation. However, its performance degrades at fast changing torque demands as e.g. for wheel slip control.

Table II: WEIGHTS OF THE DCA AND MPCA IN THE NORMAL AND EMERGENCY BRAKING SITUATION

Type	DCA $\boldsymbol{W}_1$	DCA $\boldsymbol{W}_2$	MPCA $\boldsymbol{R}$	MPCA Q_{k}
normal	$\mathrm{diag}(0, 0.025)$	$\mathrm{diag}(0, 0)$	$\mathrm{diag}(10^{-4}, 0.97)$	216
emergency	$\mathrm{diag}(6.2 \cdot 10^{-4}, 0.025)$	$\mathrm{diag}(0.074, 0.79)$	$\mathrm{diag}(0, 0.005)$	10

B. Dynamic Control Allocation (DCA)

The dynamic control allocation (DCA) approach for brake torque blending was investigated in [2]. This approach uses optimization to exploit the actuation redundancy. The goal is to find a discrete time allocation solution $\mathbf{u}_{\mathrm{k}} = [u_{\mathrm{f,k}} \quad u_{\mathrm{e,k}}]^T$ at the time sample k that allows for torque distribution in frequency domain to make the best of the actuators' different bandwidths for a fast total wheel-torque response. Moreover, actuator constraints and energy efficiency related performance metrics can be considered. The simplified mathematical formulation of this approach is described as follows,

$$\min_{\boldsymbol{u}_{\mathrm{k}}} (\|\boldsymbol{W}_1 \cdot \boldsymbol{u}_{\mathrm{k}}\|^2 + \|\boldsymbol{W}_2 \cdot (\boldsymbol{u}_{\mathrm{k}} - \boldsymbol{u}_{\mathrm{k-1}})\|^2) \tag{24}$$

$$\text{s.t.} \quad [k_{\mathrm{f}} \quad k_{\mathrm{e}}] \cdot \mathbf{u}_{\mathrm{k}} = T_{\mathrm{w}}^{*}; \quad k = 0 \tag{25}$$

$$\underline{\mathrm{u}}_{\mathrm{i}} \le \mathrm{u}_{\mathrm{i,k}} \le \overline{\mathrm{u}}_{\mathrm{i}}; \quad i \in \{\mathrm{e}, \mathrm{f}\} \tag{26}$$

where k is the discrete time sample index and $\boldsymbol{W}_1$, $\boldsymbol{W}_2$ are weighting matrices. The first constraint (25) ensures steady state achievement of the demanded wheel torque. The second constraint (26) reflects the actuator range and rate limits (8). Note that the index i stands for the type of actuator (brake or motor) and k indexes the time sample. Here, $\boldsymbol{u}_{\mathrm{k-1}}$ is the torque allocation solution obtained in the previous allocation sample.

C. Model Predictive Control Allocation (MPCA)

The Model Predictive Control Allocation (MPCA) approach [18], applied to the torque blending problem, can be conceived as an extension to DCA. The main advantage of the MPCA is related with its systematic approach to deal with the actuator dynamics. More specifically, instead of relying on an *ad-hoc* cost function to implicitly cope with the actuator dynamics (as it is done in the DCA), the MPCA explicitly incorporates the actuator's dynamic model as a constraint in the optimization problem. With this approach, more complex and precise actuator models can be handled in the torque blending process. In contrary to DCA, high-order

actuator models and actuation delays can be incorporated in the MPCA formulation.

Similar to traditional MPC, MPCA relies on optimization of a finite horizon prediction of the controlled system. (In our case, the plant is only represented by the braking actuators). The goal is to find the actuator setpoints $\mathbf{u}_k$ that will minimize a weighted cost function for trading off between traction error $\Delta T_{\mathrm{w,k}} = T_{\mathrm{w,k}} - T^*_{\mathrm{w,k}}$ and actuator energy consumption $\mathbf{u}_k$ while fulfilling the actuation constraints and dynamics. The minimization problem over the moving prediction horizon with the length N is formalized as follows:

$$\min_{\mathbf{x}_\mathrm{k},\mathbf{u}_\mathrm{k}} \quad \sum_{\mathrm{k}=0}^{\mathrm{N}} Q_\mathrm{k}\left(\Delta T_{\mathrm{w,k}}\right)^2 + \sum_{\mathrm{k}=0}^{\mathrm{N}-1} \mathbf{u}_\mathrm{k}^T \mathbf{R} \mathbf{u}_\mathrm{k} \tag{27}$$

$$s.t. \quad \mathbf{x}_{\mathrm{k}+1} = \boldsymbol{A}\boldsymbol{x}_\mathrm{k} + \boldsymbol{B}\boldsymbol{u}_\mathrm{k} \tag{28}$$

$$\Delta T_{\mathrm{w,k}} = \boldsymbol{C}\boldsymbol{x}_\mathrm{k} - T^*_{\mathrm{w,k}} \tag{29}$$

$$\underline{\mathrm{x}}_\mathrm{i} \le \mathrm{x}_{\mathrm{i,k}} \le \overline{\mathrm{x}}_\mathrm{i} \ , \ \ \underline{\mathrm{u}}_\mathrm{i} \le \mathrm{u}_{\mathrm{i,k}} \le \overline{\mathrm{u}}_\mathrm{i};\ i \in \{\mathrm{e,f}\} \tag{30}$$

The triplet $(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C})$ is the discrete time state space plant representation being composed of the discretized actuator models (7) using a first-order hold approximation. Correspondingly, $\mathbf{x}_\mathrm{k}$ is the state vector. The constraints (28)-(30) reflect the state and actuation limits (8).

The MPCA formulation (27)-(30) has three tuning parameters: N, Q_k and $\mathbf{R}$. As a rule of thumb, the control horizon N should capture the fundamental parts of the actuator transients [19]. The actuator effort is penalized by means of $\mathbf{R} = \mathrm{diag}([r_\mathrm{f} \quad r_\mathrm{e}])^T$ and the tracking error at the time k with Q_k.

The MPCA formulation adopted here follows the receding horizon strategy, well-known from MPC [19]. At each time step, (27) is solved, and then only the first element of the control vector, $\mathbf{u}[0]$, is applied to the actuators. This procedure is repeated at each sample instant.

As already mentioned, the main advantage of the MPCA lies in its superior consideration of the actuator's dynamics. In fact implicit compensation of the actuator dynamics improves the closed loop transient performance [20], [18]. On the other hand, MPCA generally involves high implementation effort and computational time to solve (27) compared to DC and DCA. Nonetheless, due to the fast real-time numerical solvers that are currently becoming available [21], together with the increasing computational capabilities of embedded systems, the real-time implementation of MPC/MPCA is getting easier to accomplish.

Table III: COMPARISON OF THE TORQUE BLENDING ALGORITHMS DURING THE SCALEDD SUPPLEMENTAL FTP DRIVING CYCLE [23]

Type of Metric	DC	DCA	MPCA
$Recup_{\mathrm{pot}}$	100%	100%	100%
$NRMS_{\mathrm{err}}$	0.2%	0.2%	0.2%
$Comp_{\mathrm{Time}}$	≈ 0	19ns	2ms

V. SUPERVISOR

A basic task of the supervisor (cf. Figure 1) is to monitor wheel slip (both traction and brake slip). In case of excessive slip the supervisor activates wheel slip control. During braking, adaptions to torque blending parameters can be scheduled by the supervisor. Notice that only the DCA and MPCA have tunable weights; for the DC no adaptation is possible. Table II presents the weights employed in this work, which varies according to the braking modes. Two modes are considered here:

1. Normal braking: the focus of this mode is to maximize energy recuperation. Thus the use of the IWM is assigned to a lower penalization; in other words, w_{11} (DCA) or r_e (MPCA) are set to a low value.
2. Emergency braking (high braking torque demand and excessive wheel slip): the focus of this mode is a fast wheel-torque response, helping to improve the dynamics of the wheel slip controller. Therefore, the weightings of the dynamic component $\mathbf{W}_2$ (DCA) or the scalar Q_k (MPCA) are emphasized.

VI. SIMULATION RESULTS

In this section the performance of the three introduced allocation strategies is compared. In order to assess the different torque blending algorithms, a predefined torque pattern is used together with the following performance metrics (adapted from [22]):

- $Recup_{\mathrm{pot}}$: potential for energy recuperation

$$Recup_{\mathrm{pot}} = \frac{-\int_0^t \min(T_\mathrm{e}, 0)\,\mathrm{d}\tau}{\int_0^t \min(|T^*_\mathrm{w}|, |\underline{T}_\mathrm{e}|)\mathrm{d}\tau} \tag{31}$$

- $NRMS_{\mathrm{err}}$: the normalized quadratic mean value of the torque error over the time t

$$NRMS_{\mathrm{err}} = \frac{\sqrt{\frac{1}{t}\int_0^t (T^*_\mathrm{w} - (T_\mathrm{f} + T_\mathrm{e}))^2 \mathrm{d}\tau}}{\left(\overline{T}_\mathrm{e} + \overline{T}_\mathrm{f}\right) - \left(\underline{T}_\mathrm{e} + \underline{T}_\mathrm{f}\right)} \tag{32}$$

- $Comp_{\mathrm{Time}}$: the mean computation time per sample evaluated using a *core i7 m6202.67GHz 8Gb RAM*

A. Driving Cycle Evaluation

The first validation test uses the supplemental FTP driving cycle [23] to generate some practical brake torque demands T^*_w as input for the torque blending. The goal of this test is to evaluate how the different torque allocation strategies perform during braking maneuvers inducing only moderate wheel slip. The aggregate results shown in Table III reveal a very similar performance among the three allocation algorithms under evaluation. In particular, all three strategies are able to maximize the energy recuperation, while keeping a low wheel-torque tracking error. The computational effort of the MPCA is increased compared to the DCA. However, the solution of the MPCA problem is still gained within milliseconds.

Table IV: COMPARISON OF THE TORQUE BLENDING ALGORITHMS WITH PRBS EXCITATION

Type of Metric	DC	DCA	MPCA
$Recup_{\text{pot}}$	100%	44%	81%
$NRMS_{\text{err}}$	23%	17%	18%
$Comp_{\text{Time}}$	≈ 0	13ns	3ms

B. Pseudo Random Binary Sequence (PRBS) Torque Demand

In the second evaluation test, a pseudo-random binary sequence (PRBS) [24] was injected in the torque setpoint T_w^* of the torque blending. The aim of this test is to evaluate how the different torque blending strategies behave when fast torque transients are demanded (which may appear as a result of the superordinate wheel slip controller activation). The PRBS torque demand is scaled such that both brake actuators become activated during the test. Figure 5 shows the result of the torque blending algorithms, when excited with the PRBS signal. Inspecting the $NRMS_{\text{err}}$, one can find that the DC is, from a torque-tracking standpoint, the worst solution. On the other hand, from an energy efficiency point of view, DC and MPCA offer significantly better recuperation capabilities than DCA. In summary, these results demonstrate that MPCA exhibits a good trade-off between tracking and energy recuperation.

Figure 5 illustrates the time-response of the three torque blending algorithms. It can be observed that both MPCA and DCA make use of the fast IWM torque and thus show good tracking of the demanded wheel torque. Furthermore, in the DCA case, the torque demand applied to the friction brake is very smooth, which is a result of the filtering properties. This stands in contrast to the MPCA's friction torque setpoint, which exhibits significant jumps after the steps of the PRBS signal. The reason for this jump is related with the predictive nature of the MPCA, which, in order to improve the tracking of the wheel torque demand, allows to pre-act on the actuator dynamics.

C. Wheel Slip Control

The last test compares the three torque blending algorithms during an emergency braking wheel slip control according to (15). The controller was applied on the four wheel driven electric vehicle in Modelica's powertrain library [25] based on the parameters presented in Table I. To make an even-handed comparison the tuning parameters of the wheel slip controller were kept fixed in all the simulations ($\omega_r = 300$, $\xi_r = 0.707$).

Figure 6 presents the response of the vehicle model when the wheel slip control is tracking a biased square wave setpoint of the wheel slip reference λ^*. As can be seen from the first subplot, the wheel slip response obtained with the DC heavily oscillates around the reference value. To better illuminate the difference between DCA and MPCA, the fourth and fifth subplots of Figure 6 show the electric and the friction torque setpoints generated during the braking maneuver. These results reveal that, despite offering similar

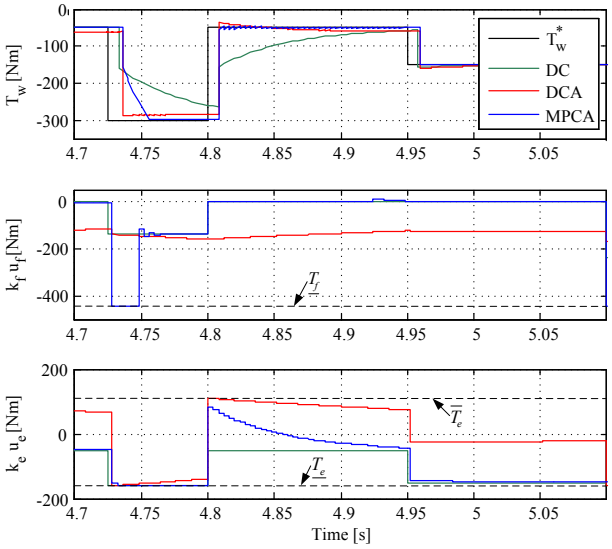

Figure 5: Comparison of the torque blending algorithms tracking a PRBS signal

total wheel braking torques, the DCA and the MPCA are using the braking actuators in a very different way. In the DCA case, the friction brakes T_f are generally varying slowly in time, while the electric motor T_e is used to cover the fast wheel-torque transients. For the MPCA case, one can find that the high bandwidth of the electric motor is still utilized when fast wheel torque setpoint steps are requested

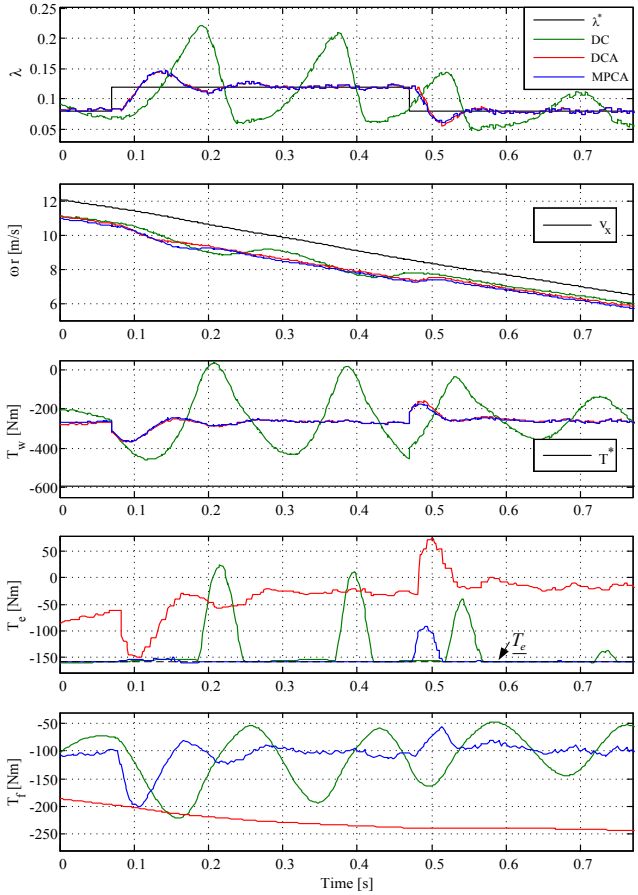

Figure 6: ABS wheel slip tracking with DC, DCA and MPCA used as torque blending strategies

(see, e.g., the transient around 0.5s). However, when the system approaches a steady-state condition, the MPCA is able to increase again the electric torque usage and maximize the energy recuperation, even during emergency wheel slip control operation. This performance degradation is due to nonlinear behavior of the DC algorithm, which is making use of the slower actuator – the FB – during the operation of the wheel slip controller.

The other two algorithms (DCA and MPCA) provide significantly better closed loop dynamics and wheel slip tracking accuracy.

VII. Conclusion & Outlook

This paper proposed a robust gain-scheduled PI wheel slip control together with a novel model predictive control allocation (MPCA) approach to solve the torque blending problem of a hybrid brake system. Additionally, we provided a detailed comparison between this MPCA and two other state-of-the-art allocation solutions (daisy chain and dynamic control allocation).

During moderate and steady braking situations, all of the *torque blending strategies provide* good results. However, whenever fast wheel torque variations are needed (which may appear as a result of the wheel slip control activation), the MPCA and dynamic control allocation outperform the daisy chain. Further, the MPCA maximizes energy recuperation, even in emergency situations, while the dynamics of the braking system is still dominated by the fast electric motor.

Future investigations will include experimental validation of both the MPCA and the wheel slip controller on our research platform, the ROMO. Another focus will be the integration of the wheel slip control in the MPCA formulation.

References

[1] Jonathan Brembeck, Lok Man Ho, Alexander Schaub, Clemens Satzger, Jakub Tobolar, "ROMO – THE ROBOTIC ELECTRIC VEHICLE," *22nd IAVSD International Symposium on Dynamics of Vehicle on Roads and Tracks*, 2011.

[2] Ricardo Castro, Rui E. Araújo, Mara Tanelli, Sergio M. Savaresi, and Diamantino Freitas, "Torque blending and wheel slip control in EVs with in-wheel motors," *Vehicle System Dynamics*, vol. 50, no. sup1, pp. 71-94, 2012.

[3] S. Torkel Glad Ola Härkegard, Resolving actuator redundancy—optimal control vs. control allocation, 2005.

[4] Ola Harkegard, "Dynamic Control Allocation Using Constrained Quadratic Programming," *AIAA Guidance, Navigation, and Control Conference*, 2002.

[5] Yu Luo, Andrea Serrani, Stephen Yurkovich, Michael W. Oppenheimer, and David B. Doman, "Model-Predictive Dynamic Control Allocation Scheme for Reentry Vehicles," *Journal of Guidance, Control, and Dynamics*, vol. 30, no. 1, pp. 100-113, 2007.

[6] Yu Luo and David B. Doman, "Dynamic Control Allocation with Asymptotic Tracking of Time-Varying Control Input Commands," *Proceedings of the American Control Conference*, 2005.

[7] Michael W. Oppenheimer and David B. Doman, "Methods for Compensating for Control Allocator and Actuator Interactions," *AIAA Guidance, Navigation, and Control Conference*, 2004.

[8] Michael W. Oppenheimer and David B. Doman, "A Method for Compensation of Interactions Between Second-Order Actuators and Control Allocators," *Aerospace Conference, IEEE*, pp. 1-8, 2005.

[9] C. Vermillion, J. Sun, and K. Butts, "Model predictive control allocation for overactuated systems - stability and performance ," in *46th IEEE Conference on Decision and Control*, New Orleans, LA , 2007.

[10] Karlo Griparic and Stjepan Bogdan Goran Vasiljevic, "Slip-based Traction Control System with an On-line Road Condition Estimation for Electric Vehicles," in *IEEE International Conference on Control Applications*, Dubrovnik, 2012, pp. 395-400.

[11] Yoichi Hori, Y Toyoda, and Y Tsuruoka, "Traction Control for Electric Vehicles," in *IEEE TRANSACTIONS ON INDUSTRY APPLICATIONS*, 1998.

[12] K. Fujii and H. Fujimoto, "Traction Control based on Slip Ratio Estimation Without Detecting Vehicle Speed for Electric Vehicle," in *Power Conversion Conference*, Nagoya, 2007, pp. pp.688,693.

[13] Martin Ringdorfer and Martin Horn, "Vehicle Dynamics Controller Concept for Electric Vehicles," in *10th International Symposium on Advanced Vehicle Control*, Leicestershire, 2010, pp. 675-680.

[14] Tilman Bünte, Lok Man Ho, Clemens Satzger, and Jonathan Brembeck, "Central Vehicle Dynamics Control of the Robotic Research Platform ROboMObil," *ATZ elektronik*, vol. 03, 2014, to appear.

[15] Sergio M. Savaresi and Mara Tanelli, *Active Braking Control Systems Design for Vehicles*,. London: Springer, 2010.

[16] Andreas Pfeiffer, "Optimization Library for Interactive Multi Criteria Optimization Tasks," *Proceedings of the 9th International Modelica Conference*, 2012.

[17] R. de Castro, R.E. Araújo, and D. Freitas, "Real-time estimation of tyre-road friction peak with optimal linear parameterisation," *Control Theory & Applications, IET*, vol. vol. 6, no. no.14, pp. pp.2257,2268, 2012.

[18] T.A. Johansen, G.K. Mykland, and A. Skullestad, "Dynamic Model Predictive Control Allocation Using CVXGEN," *The 9th IEEE International Conference on Control & Automation*, pp. 417-422, 2011.

[19] Luiping Wang, *Model Predictive Control System Design and Implementation Using Matlab*,.: Springer, 2009.

[20] Yu Luo and David B. Doman, "Model predictive dynamic control allocation with actuator dynamics," *Proceedings of the American Control Conference*, 2004.

[21] Jacob Mattingley and Stephen Boyd, "CVXGEN: a code generator for embedded convex optimization," *Optimization and Engineering*, vol. 13, no. 1, pp. 1-27, 2012.

[22] Clemens Wolfgang Satzger, Jonathan Brembeck, Martin Otter, "Framework for the Evaluation of Wheel Torque Blending Algorithms," *7th IFAC Symposium on Advances in Automotive Control,* 2013.

[23] United States Environmental Protection Agency. Dynamometer Drive Schedules Supplemental FTP driving schedule. [Online]. http://www.epa.gov/nvfel/testing/dynamometer.htm

[24] Paul H. Bardell, William H. McAnney, and Jacob Savir, *Built-In Test for VLSI: Pseudorandom Techniques*,. New York: John Wiley & Sons, 1987.

[25] Jakub Tobolar, Martin Otter, and Tilman Bünte, "Modelling of Vehicle Powertrains with the Modelica PowerTrain Library," *Systemanalyse in der Kfz-Antriebstechnik IV*, pp. 204-216, 2007.