Control Allocation for Yaw Stabilization in Automotive Vehicles using Multiparametric Nonlinear Programming

P. Tøndel[1], T. A. Johansen[1,2]

Abstract— **We investigate the use of a nonlinear control allocation scheme for automotive vehicles. Such a scheme is useful in e.g. yaw or roll stabilization of the vehicle. The control allocation allows a modularization of the control task, such that a higher level control system specifies a desired moment to work on the vehicle, while the control allocation distributes this moment among the individual wheels by commanding appropriate wheel slips. The control allocation problem is defined as a nonlinear optimization problem, to which an explicit piecewise linear approximate solution function is computed off-line. Such a solution function can computationally efficiently be implemented in real time with at most a few hundred arithmetic operations per sample. Yaw stabilization of the vehicle yaw dynamics is used as an example of use of the control allocation. Simulations show that the controller stabilizes the vehicle in an extreme manoeuvre where the vehicle yaw dynamics otherwise becomes unstable.**

I. Introduction

One of the first contributions within the field of stabilizing control of automotive vehicles was ABS brakes, which was commercially introduced more than 30 years ago. ABS controllers maintain high longitudinal braking efficiency and improve yaw stability of the vehicle during braking manoeuvres. These systems were followed by traction control systems, which improve the lateral stability and maximize friction during acceleration of the vehicle. A natural continuation of this development was to introduce automatic control of the yaw motion during critical situations to retain steerability of the vehicle over a wider operating envelope. See e.g. [1], [2], [3], [4], [5] for concepts to influence the yaw dynamics by active steering, while differential braking was used for controlling the yaw motion in e.g [6], [7]. Such control systems became commercially available in the 90's, with the Electronic Stability Program (ESP) [8]. In [9] it is shown that knowledge of the friction coefficient offers significant improvement of the vehicle response during yaw rate control.

The main objective of this paper is to present a control allocation scheme for automotive vehicles. A high level controller provides a vector of commanded forces and moments (generalized forces) which should work on the vehicle. The control allocation task is to distribute this generalized force to the individual tyres, in an optimal manner, see [10], [11]. Control allocation is a well known technique used in marine craft and aircraft. Control allocation for automotive vehicles have previously been used by [12], [13]. In [12] real-time nonlinear optimization was used to distribute the desired force and moment among the tire forces. In [13] the authors used real-time quadratic programming accompanied with a linear quadratic regulator to track a desired yaw rate trajectory while minimizing sideslip. Differential braking force at the rear and front tires and the steering angle of the front tires were used as control inputs.

By introducing control allocation into vehicle control, a large degree of modularization of the different levels of control is obtained. The control allocation routine receives a commanded generalized force to work on the vehicle from a higher level control system, e.g. a yaw stabilization system. The control allocation does in turn command a desired wheel slip, to an underlying wheel slip controller. In general, also the steering wheel and active suspension could be included, by extending the control allocation problem. This would, however, also increase the complexity of the solution. Solving the control allocation optimization problems in real-time using optimization software is not a technique applicable for practical implementation in a real vehicle, due to the high computational requirements, and complex numerical software, which is generally not recommended in safety-critical applications. A key requirement is implementation on low-cost hardware with low complexity software, which allows verification of reliability and correctness. We address this problem by designing an approximation to the optimal solution based on multiparametric nonlinear programming (mp-NLP) [14]. This is a technique in which an arbitrarily close approximation to the solution of an NLP can be found off-line as an explicit, piecewise linear (PWL) function of the state. This gives a solution which can readily be implemented and verified on inexpensive hardware with high software reliability. Multiparametric programming have previously been used in automotive applications, see [15], [16]. Moreover, in [17] multiparametric programming was used to design a lateral vehicle controller based on explicit nonlinear receding horizon control. A similar control allocation scheme based on multiparametric quadratic programming (mp-QP) was suggested for linear actuator systems in [18], and experimentally tested in ship control.

II. Vehicle Model

A three state model is used to describe the dynamics of the vehicle [19], based on the geometry in Fig. 1. We assume that only the front wheels can be steered, that is

This work was in part financed by the Norwegian Research Council, grant 157804/432, and in part by CEmACS (EU STREP project)

[1] Department of Engineering Cybernetics, Norwegian University of Science and Technology, 7491 Trondheim, Norway

[2] SINTEF Electronics and Cybernetics, 7465 Trondheim, Norway

{Petter.Tondel, Tor.Arne.Johansen}@itk.ntnu.no

0-7803-9098-9/05/$25.00 ©2005 AACC

TABLE I

NOMENCLATURE

v	Speed (absolute value of velocity vector at centre of gravity (COG))
β	Vehicle side slip angle
$\dot{\psi}$	Yaw rate
F_{xi}	Friction force on wheel in longitudinal wheel direction, $i \in \{1, 2, 3, 4\}$
F_{yi}	Friction force on wheel in lateral wheel direction, $i \in \{1, 2, 3, 4\}$
F_{zi}	Vertical force on ground from each wheel, $i \in \{1, 2, 3, 4\}$
δ_i	Steering angle, $i \in \{1, 2, 3, 4\}$
M	Total yaw moment working on vehicle
M_d	Desired yaw moment to work on vehicle
m	Vehicle mass
J	Vehicle moment of inertia about vertical axis at COG
μ_H	Maximum tyre-road friction coefficient
α_i	Wheel side slip angle (angle between velocity vector at centre of wheel and wheel direction), $i \in \{1, 2, 3, 4\}$
μ_{yi}	Lateral friction coefficient, $i \in \{1, 2, 3, 4\}$
μ_{xi}	Longitudinal friction coefficient, $i \in \{1, 2, 3, 4\}$
x, y, z	Longitudinal, lateral and vertical axes

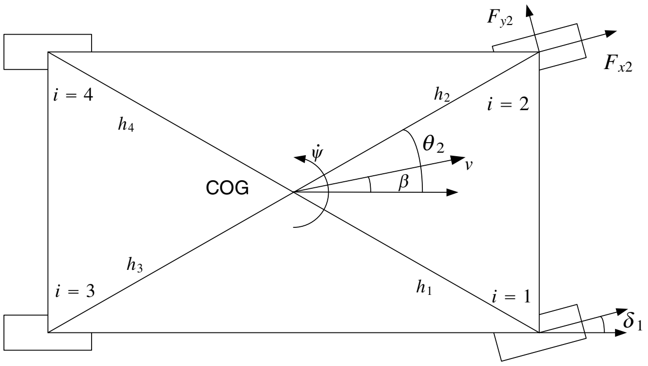

Fig. 1. Vehicle geometry and coordinate definitions

$\delta_1 = \delta_2 = \delta$, and $\delta_3 = \delta_4 = 0$. We will assume that the driver controls the front wheel steering angle δ by using the steering wheel, while the controller can use the four longitudinal wheel slips λ_{xi} for stabilizing the yaw motion. The longitudinal wheel slip during braking is defined by

$$\lambda_{xi} = \frac{v - \omega_i r}{v} \tag{1}$$

and describes the normalized difference between the vehicle speed v and the speed of the wheel perimeter $\omega_i r$. We assume that only brakes are available as control inputs, so that only negative forces F_{xi} can be generated. The wheel forces are given by the friction coefficients, which are given by nonlinear functions (friction curves) [20]:

$$F_{xi} = -F_{zi}\mu_{xi}(\lambda_{xi}, \alpha_i, \mu_H) \tag{2}$$

$$F_{yi} = F_{zi}\mu_{yi}(\lambda_{xi}, \alpha_i, \mu_H). \tag{3}$$

For simplicity we have assumed $F_{zi} = \frac{mg}{4}$. Fig. 2 shows typical friction curves for lateral forces for some values of α_i, while Fig. 3 shows a typical friction curve for longitudinal forces for $\alpha_i = 0$. The vehicle is assumed to be equipped with an ABS system with a wheel slip controller which can apply commanded longitudinal wheel slip values individually [21], [22]. From Newton's law one can derive

$$\begin{bmatrix} \dot{v} \\ \dot{\beta} \end{bmatrix} = -\begin{bmatrix} 0 \\ \dot{\psi} \end{bmatrix} + \frac{T(v, \beta)}{m}\begin{bmatrix} f_x \\ f_y \end{bmatrix}, \tag{4}$$

where

$$\begin{bmatrix} f_x \\ f_y \end{bmatrix} = \sum_{i=1}^{4} D(\delta_i)\begin{bmatrix} F_{xi} \\ F_{yi} \end{bmatrix}. \tag{5}$$

Similarly, a moment balance gives

$$\ddot{\psi} = \frac{M}{J}, \tag{6}$$

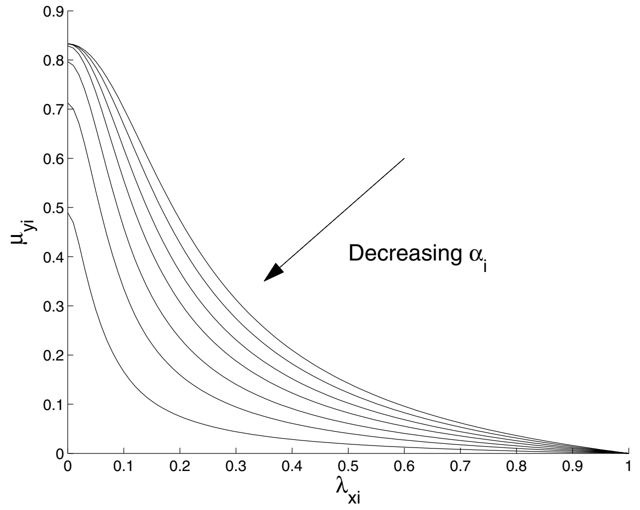

Fig. 2. Lateral friction coefficient as a function of the longitudinal wheel slip

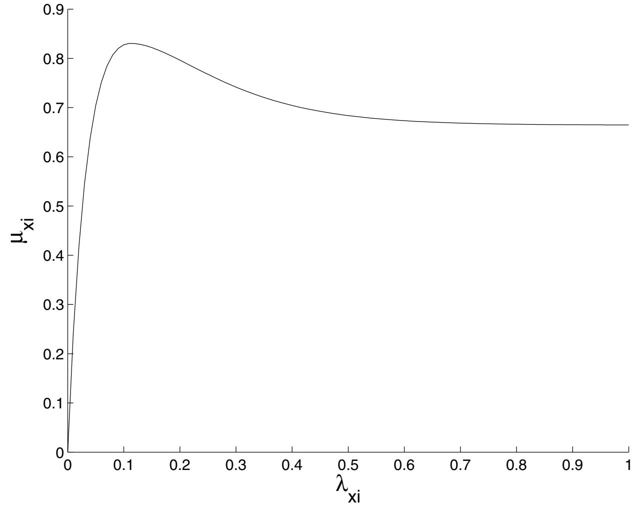

Fig. 3. Longitudinal friction coefficient as a function of the longitudinal wheel slip

where

$$M = \sum_{i=1}^{4} g_i^T(h_i, \theta_i) D(\delta_i) \begin{bmatrix} F_{xi} \\ F_{yi} \end{bmatrix}. \qquad (7)$$

Here,

$$\begin{aligned} T(v, \beta) &= \frac{1}{v} \begin{bmatrix} v\cos\beta & v\sin\beta \\ -\sin\beta & \cos\beta \end{bmatrix}, v > 0, && (8) \\ g_i(h_i, \theta_i) &= \begin{bmatrix} -\sin\theta_i \\ \cos\theta_i \end{bmatrix} h_i, && (9) \\ D(\delta_i) &= \begin{bmatrix} \cos\delta_i & -\sin\delta_i \\ \sin\delta_i & \cos\delta_i \end{bmatrix}. && (10) \end{aligned}$$

Let the state vector be $x = [\ v \quad \beta \quad \dot{\psi}\ ]^T$. We want to use feedback from the system state in the controller and control allocation, and thus need measurements/estimates of this. The vehicle speed v can be estimated from the wheel speeds and other sensors, $\dot{\psi}$ can be measured and an observer can be designed to obtain the vehicle side slip angle β [23]. We also need an estimate of the maximum friction coefficient μ_H, which we assume is available from the ABS system (see e.g. [21], [19]) or a separate system such as [24]. Moreover, the wheel side slip angles α_i are needed in the control allocation. These can be computed from the state x and the steering angle δ.

III. Yaw Moment Control Allocation Objectives

For the lateral motion of an automotive vehicle it is natural to consider the motion in the horizontal plane only, thus the relevant generalized forces to be considered are the lateral and longitudinal forces on the vehicle, f_x and f_y, and the yaw moment M around the z-axis. However, when the objective is to retain stability of the vehicle, the most important entity is the yaw moment M. Therefore we will define as the primary goal of the control allocation, to obtain a desired moment M_d using the four brakes. Introducing further constraints or objectives related to the roll moment is conceptually straightforward, but requires an extension of the model.

The control allocation scheme will be a static mapping between the commanded moment M_d, and the control inputs, without considering the dynamics of the vehicle. A higher level control system takes care of these dynamics. This means that the higher level control system may be a yaw stabilization controller that commands M_d, but it may as well be some other advanced vehicle control system, such as collision avoidance.

Given a commanded braking moment M_d from the higher level control system, the primary goal of the control allocation module is to obtain $M \approx M_d$ by commanding the appropriate wheel slips to each of the four wheels. One obviously has extra degrees of freedom with this problem formulation. In particular, it makes sense to apply as small wheel slips as possible, so we also have a secondary goal which is to find the smallest vector of wheel slips (in some sense), which fulfills the first goal. Besides the obvious requirement to use the brakes for steering purposes only when absolutely necessary, a reason for this secondary goal is that the maximal lateral force from a wheel decreases with an increasing longitudinal wheel slip λ_{xi}, and keeping a high maximal lateral force counteracts lateral skidding.

The following optimization formulation may be used to generate a vector of braking wheel slips λ_x, to obtain a desired moment to work on the vehicle:

$$\begin{aligned} \min_{\lambda_x} \quad & \frac{1}{2} b \left(M_d - \left(M\left(\lambda_x, \delta, \alpha, \mu_H\right) - M\left(0, \delta, \alpha, \mu_H\right)\right)\right)^2 + \\ & \frac{1}{2} a \lambda_x^T \lambda_x \qquad (11) \\ s.t \quad & \lambda_{xi} \geq 0, \text{ for } i \in \{1, 2, 3, 4\} \qquad (12) \end{aligned}$$

$M\left(\lambda_x, \delta, \alpha, \mu_H\right)$ is a nonlinear function given by the friction curves and (7), and a and b are constants, with b typically chosen to be relatively large compared to a, to prioritize the primary goal. Both a and b are included in the formulation to avoid numerical problems in the implementation. This formulation sees to that the brakes will not be activated for steering purposes whenever $M_d = 0$, since $M\left(0, \delta, \alpha, \mu_H\right)$ is the moment caused by driver manual steering.

IV. Yaw Stabilization

The main goal of using brake actuators to control the yaw motion, is to be able to maintain steerability of the vehicle in critical driving situations in which the driver may have difficulties or will be unable to do so by using the steering wheel alone. We will use the same strategy for maintaining vehicle steerability as suggested in [19], which is by constraining the vehicle side slip angle β and yaw rate $\dot{\psi}$ to be within upper/lower bounds. The yaw rate $\dot{\psi}$ is constrained by [19]

$$\left|\dot{\psi}\right| \leq \dot{\psi}_{\max}\left(v, \dot{v}, \beta\right) = \frac{a_{Y_{\max}} - \dot{v}\sin\beta}{v\cos\beta}, \qquad (13)$$

where $a_{Y_{\max}}$ is the maximum lateral acceleration, given by

$$a_{Y_{\max}} = \mu_{Y_{\max}} \cdot 8m/s^2, \qquad (14)$$

and $\mu_{Y_{\max}}$ is the maximum lateral friction coefficient, which in turn is a function of the maximal friction μ_H. Moreover, the vehicle side slip angle β is constrained by a function of the speed v, that is [19]

$$|\beta| \leq \beta_{\max}\left(v\right) = 10^\circ - 7^\circ \cdot \frac{v^2}{\left(40m/s\right)^2}. \qquad (15)$$

Since the simulations indicated that (15) is generally satisfied whenever (13) is enforced, we chose to leave (15) implicit in the design to reduce the complexity. Since the yaw dynamics (6) from yaw moment M to yaw rate $\dot{\psi}$ is a pure integrator a simple P-controller is used as the high level lateral stabilization controller in the simulations. The desired yaw moment $M_d = K_p e_{\dot{\psi}}$, with $K_p = 150$, and

$$e_{\dot{\psi}} = \left\{ \begin{matrix} 0, \text{ if (13) is satisfied} \\ \dot{\psi}_{\max} - \dot{\psi}, \text{ otherwise} \end{matrix} \right\}, \qquad (16)$$

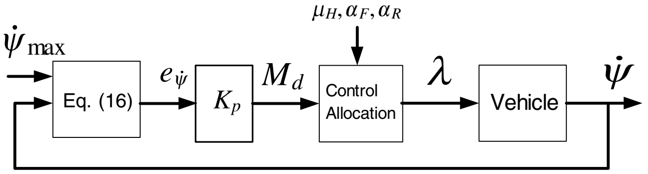

Fig. 4. Control system using yaw rate constraint and a P-controller

as seen in Fig. 4. This means that the controller is activated only when the constraint (13) is violated. An alternative to the proposed strategy of generating the desired moment M_d from the violation of constraints in the yaw rate, one may instead consider control allocation combined with a strategy as in e.g. [2], where the goal is to follow a desired yaw rate. Such a yaw rate can be generated by a prefilter from the steering wheel input and the velocity [5]. Even if this is a relatively drastic change in control objectives, the modularization offered by the control allocation would to a large extent simplify the design of such a controller.

V. Control Allocation Solution

The optimization problem (11)–(12) can be considered an mp-NLP (see Appendix), of the form (18)–(20), with $z = U$ being the variable to be optimized and $\theta = [M_d \quad \mu_H \quad \alpha_F \quad \alpha_R]^T$ being time-varying parameters to the problem, α_F and α_R are the front and rear wheel side slip angles, respectively. The explicit solution of the control allocation problem was computed in the hypercube given by

$$\begin{bmatrix} 0kNm \\ 0.1 \\ -5.7^\circ \\ -5.7^\circ \end{bmatrix} \leq \begin{bmatrix} M_d \\ \mu_H \\ \alpha_F \\ \alpha_R \end{bmatrix} \leq \begin{bmatrix} 125kNm \\ 1 \\ 5.7^\circ \\ 5.7^\circ \end{bmatrix} \tag{17}$$

Only positive M_d are considered, as the symmetry inherent in the problem can be utilized to compute the solution for negative M_d. An explicit piecewise linear approximation to the solution of (11)–(12) was generated using the algorithm [14]. The resulting partition contains 11227 polyhedral regions. By using a binary tree representation [25] of this partition, real-time evaluation of the control allocation can in the worst case be accomplished by using 58 arithmetic operations. The tree can be stored in real-time computer memory using a total of 345362 numbers. The maximal error when generating the piecewise linear approximation, was specified as $\Delta\lambda_{xi} = 0.02$. By using a Monte Carlo simulation with 100.000 random parameter values within the hybercube 17, the mean approximated error was $\Delta\lambda_m = 0.003$. As expected, the mean error is an order of magnitude less than the maximal error. The worst case approximated error found was 0.027 (see Appendix).

A few approximations were made in the design of the control allocation routine, to reduce the number of parameters in the mp-NLP, and thus the complexity of the solution.

- Even if the friction coefficient μ_H in general is different on the individual wheels, we have in the control allocation used a common μ_H on all four wheels, to reduce complexity.
- The steering angle δ is in the control allocation estimated as the difference between α_F and α_R. This approximation does in effect disregard Coriolis effects when computing δ. Actually, the wheel side slip angles α_F and α_R actually are obtained from the steering angle. However, as both α_F and α_R are needed in the control allocation to compute the friction forces, we chose to leave δ out to limit the number of parameters.
- The vertical force on each wheel is computed as $F_z = mg/4$, which means that effects from roll and pitch is disregarded. If such effects should be accounted for, this might increase the number of parameters, and thus the solution complexity.

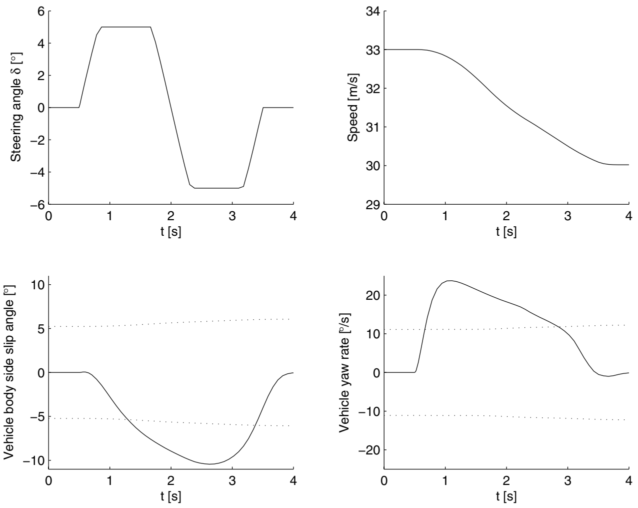

Fig. 5. Simulation of vehicle with no control action

VI. Simulations

We have considered a simulation example, in which the driver must carry out an evasive manoeuvre at high speed, with $\mu_H = 0.8$, corresponding to a dry asphalt surface. The driver controls the steering angle, and the controller should apply an appropriate distribution of braking forces, to keep the vehicle stable.

The control objective is to keep the state within the nonlinear constraints. Fig. 5 shows the uncontrolled behavior of the system. The vehicle is obviously unable to follow the drivers command. The yaw rate increases in the left turn, but the vehicle does not respond when the driver makes the right turn. The dotted lines in the β- and $\dot{\psi}$-plots are the constraints (13), (15). Fig. 6 shows a simulation of the vehicle with the controller applied. In the first turn the controller applies a braking force to both right wheels to keep the yaw rate within its bounds. A production ESP controller might in such situations apply a force to the front wheel only. This is due to the fact that F_{yi} decreases when a wheel is braking (see Fig. 2), and one wants to keep a high maximal lateral force on the rear wheels. However, the

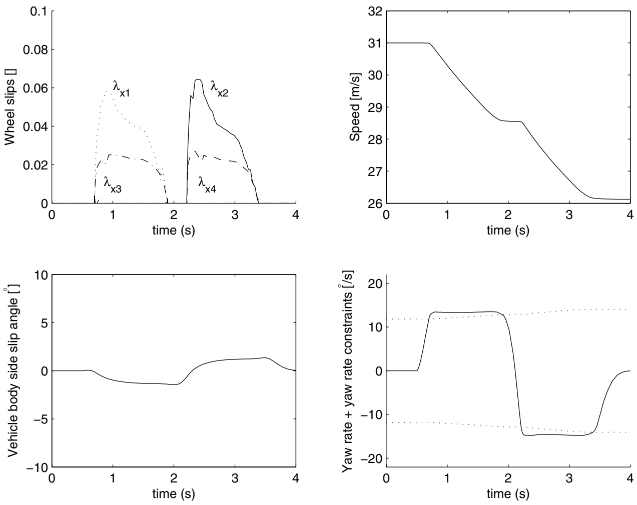

Fig. 6. Simulation of vehicle with yaw stabilization/control allocation

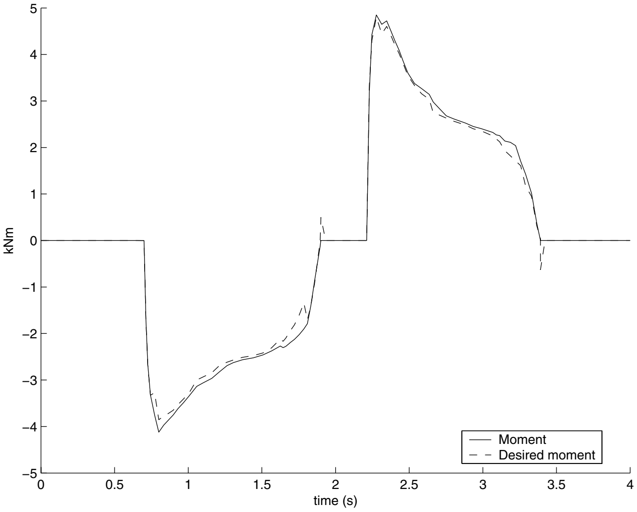

Fig. 7. Desired and actual moment

optimal (in the sense of the previously defined optimization problem) control action is to apply a small force to the rear wheel also. Since only a small wheel slip is requested on the rear wheels, most of the lateral force F_{yi} is retained, while the longitudinal force F_{xi} contributes to keeping control of the yaw dynamics. In the second turn, a similar braking torque is applied to the left wheels. As can be seen in Fig. 7 the control allocation does in this case let the moment M track the desired moment M_d relatively accurately.

VII. Conclusions

We have presented a scheme for automotive control allocation, based on solving nonlinear optimization problems. An explicit approximate solution to the optimization problems was computed, giving a solution with favorable properties for real-time implementation in a vehicle.

The control allocation function was applied in a lateral stabilization controller. Simulations have been made on a case in which the vehicle loses steerability under manual control with steering wheel only, but with the controller applied, the manoeuvre remains stable. A simple P-controller was used for keeping constraints on the yaw rate, providing stabilization of the yaw motion. However, the modularization offered by the control allocation allows the control objective to be changed into e.g. following a yaw rate reference, without altering the control allocation routine.

References

[1] J. Ackermann, J. Guldner, W. Sienel, R. Steinhauser, and V. Utkin, "Linear and nonlinear controller design for robust automatic steering," *IEEE Transactions on Control Systems Technology*, vol. 3, no. 1, pp. 132–143, 1995.

[2] J. Ackermann, D. Odenthal, and T. Bünte, "Advantages of active steering for vehicle dynamics control," in *32nd International Symposium on Automotive Technology and Automation*, Vienna, 1999, pp. 263–270.

[3] S. Mammar and D. Koenig, "Vehicle handling improvement by active steering," *Vehicle System Dynamics*, vol. 38, no. 3, pp. 211–242, 2002.

[4] J. Ackermann and T. Bünte, "Yaw disturbance attenuation by robust decoupling of car steering," *Control Eng. Practice*, vol. 5, no. 8, pp. 1131–1136, 1997.

[5] ——, "Handling improvement of robust car steering," in *International Conference on Advances in Vehicle Control and Safety*, Amiens, France, 1998.

[6] A. T. van Zanten, R. Erhardt, and G. Pfaff, "VDC, the vehicle dynamics control system of BOSCH," *SAE 950759*, 1995.

[7] V. Alberti and E. Babbel, "Improved driving stability of active braking of the individual wheels," in *Proc. of the International Symposium on Advanced Vehicle Control*, 1996, pp. 717–732.

[8] H. Bauer, Ed., *ESP Electronic Stability Program*, 1st ed. Stuttgart: Robert Bosch GmbH, 1999.

[9] T. Shim and D. Margolis, "Using μ feedforward for vehicle stability enhancement," *Vehicle System Dynamics*, vol. 35, no. 2, pp. 103–119, 2001.

[10] W. C. Durham, "Constrained control allocation," *J. Guidance, Control and Dynamics*, vol. 16, pp. 717–725, 1993.

[11] M. Bodson, "Evaluation of optimization methods for control allocation," *J. Guidance, Control and Dynamics*, vol. 25, pp. 703–711, 2002.

[12] Y. Hattori, K. Koibuchi, and T. Yokoyama, "Force and moment control with nonlinear optimum distribution for vehicle dynamics," in *Proc. of the 6th International Symposium on Advanced Vehicle Control*, Hiroshima, 2002, pp. 595–600.

[13] J. H. Plumlee, D. M. Bevly, and A. S. Hodel, "Control of a ground vehicle using quadratic programming based control allocation techniques," in *Proc. American Contr. Conf.*, Boston, 2004, pp. 4704–4709.

[14] T. A. Johansen, "On multi-parametric nonlinear programming and explicit nonlinear model predictive control," in *Proc. 41st IEEE Conf. on Decision and Control*, vol. 3, Las Vegas, NV, 2002, pp. 2768–2773.

[15] A. Bemporad, F. Borrelli, L. Glielmo, and F. Vasca, "Optimal piecewise-linear control of dry clutch engagement," in *3rd IFAC Workshop: Advances in Automotive Control*, Karlsruhe, 2001.

[16] F. Borrelli, A. Bemporad, M. Fodor, and D. Hrovat, "A hybrid approach to traction control," in *Proc. 4th International Workshop on Hybrid Systems: Comp. and Control*, Rome, 2001.

[17] P. Tøndel and T. A. Johansen, "Lateral vehicle stabilization using constrained nonlinear control," in *Proc. European Contr. Conf.*, Cambridge, 2003.

[18] T. A. Johansen, T. I. Fossen, and P. Tøndel, "Efficient optimal constrained control allocation via multi-parametric programming," *J. Guidance, Control and Dynamics*, 2005, accepted for publication.

[19] U. Kiencke and L. Nielsen, *Automotive Control Systems*. Springer-Verlag, 2000.

[20] H. B. Pacejka, E. Bakker, and L. Lidner, "A new tire model with application in vehicle dynamics studies," *SAE 890087*, 1989.

[21] M. Burckhardt, *Fahrwerktechnik: Radschlupf-Regelsysteme*. Würzburg: Vogel Verlag, 1993.

[22] T. A. Johansen, I. Petersen, J. Kalkkuhl, and J. Ludemann, "Gain-scheduled wheel slip control in automotive brake systems," *IEEE Trans. Control Systems Technology*, vol. 11, pp. 799–811, 2003.

[23] Y. Fukada, "Slip-angle estimation for vehicle stability control," *Vehicle System Dynamics*, vol. 32, no. 4-5, pp. 375–388, 1999.

[24] F. Gustafsson, "Slip-based tire-road friction estimation," *Automatica*, vol. 33, pp. 1087–1099, 1997.

[25] P. Tøndel, T. A. Johansen, and A. Bemporad, "Computation of piecewise affine control via binary search tree," *Automatica*, vol. 39, no. 5, pp. 945–950, 2003.

[26] A. Bemporad, M. Morari, V. Dua, and E. N. Pistikopoulos, "The explicit linear quadratic regulator for constrained systems," *Automatica*, vol. 38, no. 1, pp. 3–20, 2002.

[27] P. Tøndel, T. A. Johansen, and A. Bemporad, "An algorithm for multi-parametric quadratic programming and explicit MPC solutions," *Automatica*, vol. 39, no. 3, pp. 489–497, 2003.

[28] P. E. Gill, W. Murray, M. A. Saunders, and M. H. Wright, "Users guide for NPSOL 5.0: A fortran package for nonlinear programming," Systems Optimization Laboratory, Department of Operations Research, Stanford University, Tech. Rep. SOL 86-2, 1998.

Appendix. Approximate Explicit Solutions to the Multiparametric Nonlinear Programming Problems

To obtain a solution to the optimization problems from Section IV which is well suited for real-time implementation, we find an approximate solution in the form of a PWL function by using multiparametric nonlinear programming (mp-NLP) [14]. The NLPs described in Section IV can be written in the general form,

$$\min_z J(z,\theta) \tag{18}$$

$$G(z,\theta) = 0 \tag{19}$$

$$H(z,\theta) \leq 0, \tag{20}$$

where $z \in \mathbb{R}^m$ is a vector of decision variables, while $\theta \in \mathbb{R}^p$ is to be considered a parameter vector to the optimization problem. (18)–(20) are to be considered an mp-NLP, which means that the optimal solution is to be found for a given range of the parameter values. We seek to find an explicit approximate representation of the solution as a function of these parameters. A general algorithm to give an exact solution on explicit form for this problem does not exist, however, an approximate solution can be found by using the algorithm from [14], which does this by partitioning the parameter space into a set of hypercubes, and giving a quadratic approximation of the NLP in each hypercube. Each of these approximations can be treated as a multiparametric quadratic program (mp-QP) which can be solved exactly off-line, giving the solution as an explicit PWL function of its time-varying parameters [26]. A brief summary of the method for solving mp-NLPs follows (see [14] for details).

1) Let Θ be a hypercube of parameters θ in which the mp-NLP is to be solved. Let $\Theta_0 = \Theta$.
2) Select θ_0 as the centre point of Θ_0.
3) Solve the NLP (18)–(20) with $\theta = \theta_0$ by using an NLP solver to obtain the solution $z^*(\theta_0)$.
4) Compute a local quadratic approximation to the NLP cost function, and a linear approximation to the NLP constraints G and H at $(\theta_0, z^*(\theta_0))$. Such an approximation defines an mp-QP.
5) Estimate the maximal errors (in cost function, control inputs and constraint violations, see [14] for details) made by approximating the mp-NLP solution with the mp-QP solution for $\theta \in \Theta_0$. If these are not within apriori defined bounds, sub-partition Θ_0 into two new hypercubes.
6) Select a new Θ_0 from the partition. If no further sub-partitioning is needed, go to step 7. Otherwise, go to Step 2.
7) For every sub-partition Θ_0, solve the local mp-QP problem to get an exact PWL solution to the approximate mp-QP given by the function $\hat{z}_0 : \Theta_0 \rightarrow \mathbb{R}^m$.

Taking these mp-QP solutions together give an approximate solution of the NLP as a function of the parameter, $\hat{z} : \Theta \rightarrow \mathbb{R}^m$. This means that the main effort of solving the optimization problem is moved off-line. Efficient off-line mp-QP solvers have recently been developed [27]. The on-line effort is reduced to evaluating a PWL function, which can be implemented efficiently using a binary search tree as in [25]. Such a binary tree representation is equivalent to representing the polyhedral partition by the set of polyhedral regions, but is optimized for improved real-time performance. This search tree must in real-time be traversed once for each sample, to obtain the solution to the optimization problem. Such a representation allow us to apriori state the worst number of arithmetic operations needed to evaluate the control law. Moreover, the calculation is non-recursive and fixed-point arithmetic is sufficient.

The complexity of the explicit solution, in terms of real-time computational and memory requirements, does to a high degree depend on the number of parameters and constraints in the mp-NLP. The choice of parameters is therefore an important decision in the design process. The solution complexity does also depend upon the chosen maximal error in the solution, cost function and constraints.

Note that larger errors in the explicit solution than specified may appear for the following reasons:

- If the hypercubes found when doing the mp-NLP partitioning are smaller than the accuracy of the measurements/observer errors in the corresponding parameter, further subdivision of the hypercubes is avoided, to reduce solution complexity.
- The "maximal" errors in step 5 may be incorrect for nonconvex problems, and there may exist parameters for which the error is larger than specified.

Due to the non-convexity of the optimization problems, care must be taken to ensure that the global optimizer is obtained. This is another reason for using an explicit solution approach, as this gives the possibility of off-line verification of the solution. The off-line nonlinear optimization problems were solved by *NPSOL* [28].