Generation of continuously adjustable vehicle models using symbolic reduction methods

Lars Mikelsons, Thorsten Brandt

Angaben zur Veröffentlichung / Publication details:

Mikelsons, Lars, and Thorsten Brandt. 2011. "Generation of continuously adjustable vehicle models using symbolic reduction methods." *Multibody System Dynamics* 26 (2): 153–73. https://doi.org/10.1007/s11044-011-9247-x.

Generation of continuously adjustable vehicle models using symbolic reduction methods

Lars Mikelsons · Thorsten Brandt

Abstract Vehicle dynamics has been an active field of research for many decades. The well-known single-track model already introduced in 1940 by Riekert and Schunck is still used to explain fundamental effects in vehicle dynamics such as under- and oversteering. However, meanwhile also very complex multibody dynamics models exist, which allow very detailed simulations. On the other hand, real-time computations necessary for active safety and driver assistance systems are demanded for models of lower complexity. As important effects at the handling limits are not covered by the linear single-track model, but complex multibody models cannot be integrated fast enough by electronic control units of safety systems, models with an adjustable degree of complexity are desired. As an example, model-predictive control is an upcoming field of application and relies on models that are integrated over the prediction horizon. Therefore, the selection of an appropriate model is an important task. The crux of the matter is to make a compromise between computation time and accuracy of the model having only a rough guess of the accuracy of the model. In this contribution a systematic approach is proposed. Instead of selecting an existing model, which is supposed to match requirements in computation time and level of detail, a complex model is reduced to match the requirements using symbolic model reduction techniques. This approach has two major advantages: First, the accuracy can be set by the user. Second, the model is continuously adjustable in its complexity or propagation precision.

Keywords Vehicle models · Model reduction · Symbolic reduction techniques · Real time · Assistance systems

1 Introduction

Vehicle models are essential for detailed simulation purposes as well as for model-based control methods applied in modern driver assistance systems. Hence, in the last decades

L. Mikelsons (✉) · T. Brandt
Institute for Mechatronics and System Dynamics, University of Duisburg-Essen, Essen, Germany
e-mail: mikelsons@imech.de

T. Brandt
e-mail: brandt@imech.de

much effort was put into the development of adequate vehicle models. Although, for these two application examples different models are demanded. Certainly, for a detailed simulation a much more complex model will be employed than in the control algorithm of a driver assistance system which has to run in real-time. Thus, on the one hand simple models like the single-track model introduced by Riekert and Schunck [10] and on the other hand complex multibody vehicle models [9] can be found in literature. While the simpler models cover only fundamental effects like over- and understeering, complex models describe vehicle dynamics very detailed. In-between models reproducing only certain effects in detail can be found. All of these models have one thing in common: They are designed manually, depending on the experience and intuition of an engineer. Therefore, this engineer has to decide which physical effects are taken into account. This choice depends on the estimated influence on the vehicle dynamics and the available computing power. Especially for the upcoming field of model-predictive control, the selection of an appropriate model is a hard task since the models are integrated over the prediction horizon. Due to the limited computation power of an electrical control unit, the employed model has to be simple enough to allow for real-time control. Thus, up to now only look-up tables are employed, instead of physical models. However, upcoming systems like lane keeping assistance use model based control strategies, which require real-time simulations. In this contribution, a systematic approach for the generation of models of appropriate complexity is proposed. As opposed to the classical approach, it is started from a complex model. The reduced model is obtained from the complex model by the application of symbolic reduction techniques as presented in [14]. As a major advantage the reduced model is guaranteed to stay within user defined error bounds (at least for the reduction scenario, as will be shown in the next section). Symbolic reduction techniques were first used in analog circuit design [1] and based on the DC-analysis of nonlinear analog circuits. These techniques were extended to the reduction of arbitrary DAE-systems in [13, 14]. Hence, symbolic reduction techniques can be used for the modeling and design of mechatronic systems [11].

This contribution is structured as follows. In Sect. 2, symbolic reduction techniques are introduced. Moreover, a new symmetry reduction is presented. Afterwards the vehicle models which are chosen here as starting models for the reduction process are discussed. These models are on the one hand a spatial multibody vehicle model and on the other hand a nonlinear single-track model. In Sect. 4, first the reduction algorithm is explained using the rather simple single-track model as an example. Finally, the complex multibody model is reduced for different scenarios (initial values, system inputs, parameters). The paper closes with an outlook and a conclusion in Sect. 5.

2 Symbolic reduction techniques

Many model reduction approaches try to identify slow and fast states in order to project the system (in general a DAE-system) onto the manifold of the slow states [5] or to approximate the fast states [3]. Symbolic reduction techniques do not operate on the states of the mathematical model but on single terms. Hence, instead of identifying fast states, those manipulations on terms of the system are identified which have only a small influence on the simulation result. In this context such a manipulation of a term is called a reduction. Different reduction techniques are for example the cancellation of terms, the substitution of terms by constants and the linearization of terms. Hence, a reduction is the manipulation of a certain term by a certain reduction technique. An exception is the symmetry reduction, which is explained later on. A procedure which measures the influence of those reductions

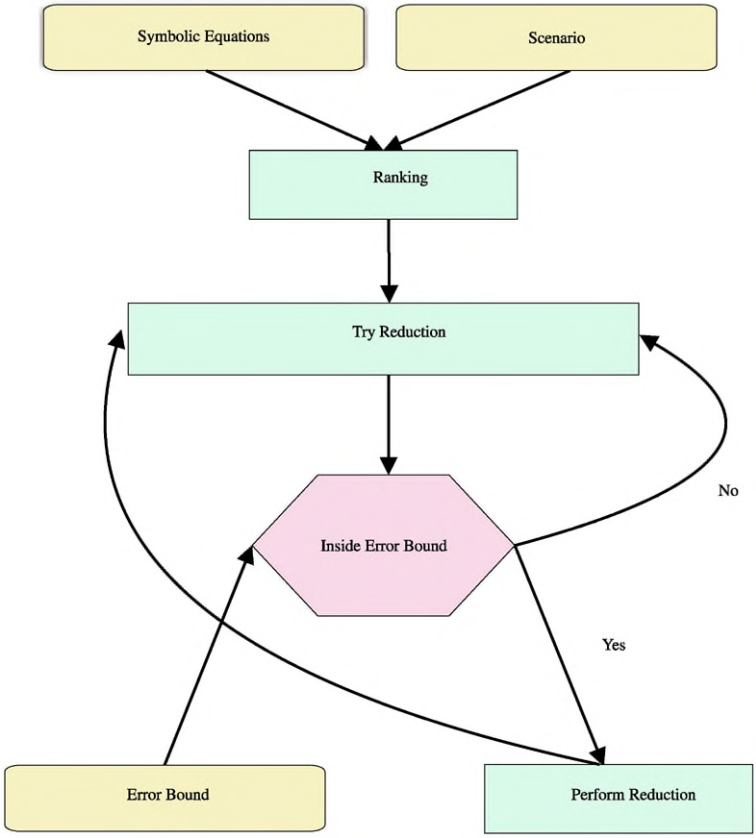


Fig. 1 Scheme of the reduction algorithm

on the solution of the system is called ranking [14]. Two different ranking procedures will be discussed in Sect. 2.2. After the ranking is computed, the terms are sorted in ascending order with respect to their influence on the solution in order to perform the reductions as long as the solution of the reduced DAE-system remains within a user-defined error bound $\boldsymbol{\varepsilon}$.

A scheme of the symbolic reduction algorithm is shown in Fig. 1 for a chosen reduction technique. Given a scenario (system inputs, initial states and parameters) and an error bound, the algorithm starts with the ranking. Here, the influence of all reductions for a chosen reduction technique onto the solution of the system is estimated. Afterwards it is checked whether the reductions lead to an error inside the error bounds, beginning with the smallest. Finally a less detailed model is obtained within the prescribed error bounds results. These steps are further explained in the next sections. First, the used nomenclature is introduced. Let

$$\mathbf{F} : \Omega \times I \mapsto \mathbb{R}^m \tag{1}$$

be differentiable, where $\Omega \subset \mathbb{R}^n \times \mathbb{R}^n$ is an open set and $I \subset \mathbb{R}$ is an interval. Then

$$\mathbf{F}(\mathbf{x}, \dot{\mathbf{x}}, t) = 0 \tag{2}$$

is called DAE-system if $\frac{\partial \mathbf{F}}{\partial \dot{\mathbf{x}}}$ is singular. Furthermore, let $\mathbf{F}$ be given in expanded form

$$\mathbf{F}_i(\mathbf{x}, \dot{\mathbf{x}}, t) = \sum_{k=1}^{l_i} t_{k_i}(\mathbf{x}, \dot{\mathbf{x}}, t), \quad 1 \leq i \leq m, \tag{3}$$

where l_i is the number of terms in $\mathbf{F}_i$ and t_{k_i} denotes the kth term in $\mathbf{F}_i$. Then, each term t_{k_i} may again be a function whose argument is a sum of subterms (terms of the second level), and so on. Consequently, the reduction can be performed on different levels of the DAE-system. Then the set $\mathcal{T}^i$ is the set of all terms in the ith level and $\mathcal{K}^i$ denotes the set of all reductions for one reduction technique in the ith level. For $\kappa \in \mathcal{K}$,

$$\mathbf{F}^\kappa = 0 \tag{4}$$

is the DAE-system emerging from the reduction κ. Then for DAE-systems of the form of (2),

$$\mathbf{F}(\mathbf{x}, \dot{\mathbf{x}}, t, \mathbf{u}) = 0 \tag{5}$$

with system inputs $\mathbf{u}$, a scenario is the set of a vector field defined on the interval I for the system inputs, the initial values and the parameters. Furthermore, $\mathcal{N}(\mathbf{F}(\mathbf{x}, \dot{\mathbf{x}}, t), \mathbf{u})$ is the solution of (2) computed by a numerical integrator $\mathcal{N}$ at nodes $t_1, \ldots, t_N$. The solution

$$\mathbf{y} = \begin{bmatrix} \mathbf{y}_{\text{out}} \\ \bar{\mathbf{y}} \end{bmatrix} = \mathcal{N}\big(F(\mathbf{x}, \dot{\mathbf{x}}, t), \mathbf{u}\big) \tag{6}$$

consists of two components. In $\mathbf{y}_{\text{out}}$ the n_{out} output variables are contained, while $\bar{\mathbf{y}}$ consists of the remaining internal variables.

2.1 Reduction techniques

As already mentioned above, possible reduction techniques are the negligence of terms ($\mathcal{U}_{\text{neg}}$), setting terms to constants ($\mathcal{U}_{\text{const}}$), linearization of terms ($\mathcal{U}_{\text{lin}}$), symmetry considerations ($\mathcal{U}_{\text{sym}}$) and the simplification of piecewise functions. In this contribution, $\mathcal{U}_{\text{neg}}$, $\mathcal{U}_{\text{const}}$, $\mathcal{U}_{\text{lin}}$ and $\mathcal{U}_{\text{sym}}$ are chosen in different compositions. Certainly, the easiest manipulation of a term is its negligence. As shown in (3), each term consists of a function. Choosing linearization as reduction technique, the term (function) under consideration is linearized with respect to its arguments. Using $\mathcal{U}_{\text{const}}$ for each term, a constant has to be chosen. Usually the mean value throughout the simulation is employed. Clearly, this mean has to be determined in advance. At first sight, this looks like a drawback, but a reference simulation is essential for the ranking anyway as will be seen in the next section. However, other values than the mean value are thinkable. Choosing $\mathcal{U}_{\text{sym}}$, at first variables which have similar values throughout the simulation are sought. Alternatively, variables which are expected to be similar can be flagged. For two similar variables, every occurrence of the first variable (or its derivative) is substituted by the second variable (or its derivative). Consequently, now one equation can be canceled. A reasonable choice is that equation which leads to smallest error.

2.2 Ranking

In general, a ranking procedure estimates the influence of a reduction on the solution of a DAE (or ODE) system. In [13], different ranking algorithms are proposed. A reasonable measure for the influence of a reduction is the error emerging from the reduction. In order to

get a good estimate of that error, a reference solution $\mathbf{y}^\star$ is required. The crux of the matter is that the quality of the estimate raises with the duration of the ranking procedure. Hence, a ranking procedure should be a good compromise between computation time and accuracy. Mathematically speaking, a ranking procedure $\mathcal{R}$ maps two DAE-systems on a real value, estimating the error between their solutions. Apparently, perfect accuracy can be achieved by the use of simulations. However, this would lead to very high computation times. Here, the so-called Residual Ranking and One-Step Ranking will be discussed.

2.2.1 Residual Ranking

Let $\mathbf{y}^\star$ be the reference solution calculated by a numerical integrator $\mathcal{N}$

$$\mathbf{y}^\star = \mathcal{N}(\mathbf{F}). \tag{7}$$

Then

$$\|\mathbf{F}(\mathbf{y}^\star(t_i), \dot{\mathbf{y}}^\star(t_i), \mathbf{u}(t_i))\| < \eta, \quad (1 \le i \le N) \tag{8}$$

holds for a prescribed accuracy η. Hence,

$$\mathcal{R}_{\text{res}}(\mathbf{F}, \kappa) = \left\| \begin{bmatrix} \|\mathbf{F}^\kappa(\mathbf{y}^\star(t_1), \dot{\mathbf{y}}^\star(t_1), \mathbf{u}(t_1))\| \\ \vdots \\ \|\mathbf{F}^\kappa(\mathbf{y}^\star(t_N), \dot{\mathbf{y}}^\star(t_N), \mathbf{u}(t_N)\|) \end{bmatrix} \right\| \tag{9}$$

seems to be a good estimate for the error resulting from the reduction $\kappa \in \mathcal{K}$. The Residual Ranking is a very fast ranking procedure due to its simplicity.

2.2.2 One-Step Ranking

Typically, computing the solution of a DAE-system, at each time step a nonlinear system of equations is iteratively solved. DASSL, for example, uses the backward difference formulae (BDF) up to order 5 in order to transform (2) into the nonlinear system of equations

$$\mathbf{F}\left(\frac{1}{\beta h}\sum_{k=0}^{i} \alpha_k \mathbf{x}_{n+1-k}, \mathbf{x}_{n+1}, t_{n+1}\right) = \mathbf{0}, \tag{10}$$

where α and β are the coefficients of the ith order BDF. Furthermore, h denotes the step-size and $\mathbf{x}_n$ is the solution at t_n. This nonlinear system of equations is solved using Newton's method. The required initial guess is obtained by an extrapolation of the solution of the preceding time-step. The idea of the One-Step Ranking is to use the reference solution for an estimation of the difference of the solution of the original and the reduced DAE-systems. Hence, during the solution of (4), the reference solution $\mathbf{y}^\star$ is used as the initial guess in Newton's method at the corresponding time steps. Since not the exact solution but an estimation of the solution of the reduced DAE is required, the number of iterations is limited to one. This gives the approximate solution $\widehat{\mathbf{y}}$ which leads to a ranking value

$$\mathcal{R}_{\text{step}}(\mathbf{F}, \kappa) = \|\mathbf{y}^\star_{\text{out}} - \widehat{\mathbf{y}}_{\text{out}}\|. \tag{11}$$

The One-Step Ranking is more accurate than the Residual Ranking and delivers a good compromise between accuracy and run-time.

2.3 Reduction algorithm

In the term-cancellation procedure, the ranking is used to perform as many reductions as possible, while preserving the desired accuracy. Here, the error emerging from the reductions is measured only at the n_{out} output variables. Thus, the error bound $\boldsymbol{\varepsilon}$ has dimension n_{out}. To perform as many reductions as possible, it is beneficial to start with those reductions which are supposed to lead only to a small error. After choosing a set of reductions $\mathcal{K}$ (by choosing a reduction technique and a level), the ranking list $\mathcal{L}$ is computed, i.e. the ranking values for all $\kappa \in \mathcal{K}$ are calculated and stored in $\mathcal{L}$ together with the corresponding reductions. Then, the ranking list $\mathcal{L}$ is sorted in ascending order with respect to the ranking value, resulting in $\mathcal{L}_{\text{sort}}$. Now, one possibility is to check the reductions in the same order as they are stored in $\mathcal{L}_{\text{sort}}$. After each reduction, it has to be verified that the solution of the reduced system $\mathbf{y}^{\kappa}$ stays within the error bound $\boldsymbol{\varepsilon}$, i.e.

$$\left\| y^{\star}_{\text{out,i}} - y^{\kappa}_{\text{out,i}} \right\|_{\infty} < \varepsilon_i \quad \forall i \leq n_{\text{out}}. \tag{12}$$

If (12) is not satisfied, κ is retracted. For this verification a simulation has to be performed. Thus, in case of complex systems, many simulations are necessary. However, this method can be accelerated by the use of clusters [12]. Using clusters, the ranking list $\mathcal{L}_{\text{sort}}$ is divided into s disjunct subsets

$$\mathcal{L}_{\text{sort}} = \bigcup_{i=1}^{s} \mathcal{S}_i, \tag{13}$$

where

$$\mathcal{S} = [S_1, \ldots, S_s]. \tag{14}$$

Each cluster $\mathcal{S}_i$ contains reductions leading to a similar estimated error (for example, up to a factor of 10). Now the clusters are checked one after another, beginning with S_1 containing the reductions leading to the smallest estimated error. Thus, multiple reductions can be verified by one simulation. If a cluster $\mathcal{S}_i$ cannot be verified (the reductions of $\mathcal{S}_i$ lead to errors greater than the error bound $\boldsymbol{\varepsilon}$), $\mathcal{S}_i$ is divided disjunct into two clusters $\mathcal{S}_i^1$ and $\mathcal{S}_i^2$. The term-cancellation procedure then continues with $\mathcal{S}_i^k$ $(k = 1, 2)$. The whole reduction algorithm is shown in Algorithm 1 for a reduction technique $\mathcal{U}$, a ranking procedure $\mathcal{R}$, a numerical integrator $\mathcal{N}$ and a certain level k. Here, for a reduction $\kappa \in \mathcal{K}$, κ^{-1} undoes the reduction. There exist different criteria to stop the algorithm. One possibility is to check only those reductions which have a ranking value lower than a number $r^{\max}$. However, in general it is hard to identify a reasonable value for $r^{\max}$ a priori. Another possibility is to stop the algorithm after $n_{\text{fail}}^{\max}$ failed reductions. Throughout this paper, the second stopping criterion with $n_{\text{fail}}^{\max} = 15$ for the spatial multibody vehicle model in Sect. 4.2 and $n_{\text{fail}}^{\max} = 3$ for the single-track model in Sect. 4.1 is employed. It should be noted that the efficiency of this algorithm depends strongly on the order of the reductions and therefore on the employed ranking procedure.

Algorithm 1 REDUCTION ALGORITHM

Require: DAE-system $\mathbf{F}$, Error Bound $\boldsymbol{\varepsilon}$, Scenario $\mathbf{u}$
Ensure: Reduced DAE-system $\mathbf{G}$
  $y_{\text{out}}^{\text{ref}} = \mathcal{N}(\mathbf{F}, \mathbf{u})$
  $\mathbf{G} = \mathbf{F}$
  $r = \mathcal{R}(\mathbf{F}, \mathcal{K})$
  $n_{\text{fail}} = 0$
  Compute $\mathcal{S}$ from r
  while $\mathcal{S} \neq \emptyset$ **and** $n_{\text{fail}} < n_{\text{fail}}^{\max}$ **do**
    $\mathcal{S}_t = \mathcal{S}_1$
    $\mathcal{S} = \mathcal{S} \setminus \mathcal{S}_1$
    for all $\kappa \in \mathcal{S}_t$ **do**
      $\mathbf{G} = \mathbf{G}^{\kappa}$
    end for
    $\mathbf{y}_{\text{out}} = \mathcal{N}(\mathbf{G}, \mathbf{u})$
    $\boldsymbol{\varepsilon}_{t_i} \Leftarrow \| y_{\text{out}_i}^{\text{ref}} - \mathbf{y}_{\text{out}_i} \|, 1 \leq i \leq n_{\text{out}}$
    if $\boldsymbol{\varepsilon}_t \geq \boldsymbol{\varepsilon}$ **then**
      for all κ_i **do**
        $\mathbf{G} = \mathbf{G}^{\kappa^{-1}}$
        $n_{\text{fail}} = n_{\text{fail}} + 1$
      end for
      if $|\mathcal{S}_t| > 1$ **then**
        Divide $\mathcal{S}_t$ **into** $\mathcal{S}_t^1$ **and** $\mathcal{S}_t^2$
        $\mathcal{S} = [\mathcal{S}_t^1, \mathcal{S}_t^2, \mathcal{S}]$
      end if
    end if
  end while

3 Vehicle models

3.1 Description of the spatial twin track model

The spatial twin track model covers all six rigid body d.o.f. of the vehicle chassis. The translational d.o.f. consist of the jerk, the lateral push and the lift. The three rotational d.o.f. (roll, pitch, yaw) are denoted as θ, φ and ψ. Additionally the model has four independent wheels. Each wheel has a rotational d.o.f. about the wheels axis and a translational d.o.f. in the vertical direction. The front wheels perform the steering motion. Altogether the twin track model has 14 d.o.f. The wheels are connected to the chassis with spring- and damper elements. Moreover, anti-roll bars in the front and the rear of the vehicle are considered. The chassis and each wheel have corresponding masses and inertia. Change of camber and caster is neglected at the wheel suspension. The inputs of the model are the steering angle and the driving torque. The driving torque acts only at the front axle and is distributed equally to the front tires.

3.1.1 Equation of motion of the twin track chassis

The quantities and parameters describing a twin track model in the x–y plane are shown in Fig. 2. Here l_F is the distance between front axle and the center of gravity (CG) and l_R is

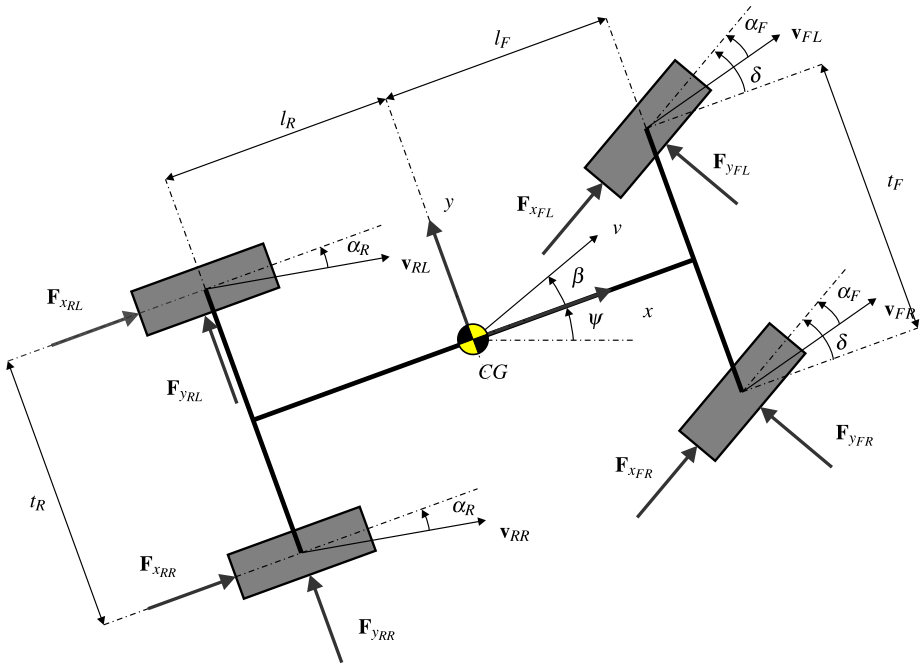


Fig. 2 Twin track model in the x–y plane

the distance between rear axle and CG. Furthermore, t_F and t_R are the track widths of the front and rear wheels, respectively. Important quantities for the lateral dynamics are the yaw angle ψ and the side slip angle β, the latter describing the angle between longitudinal axis of the vehicle and the velocity vector in the x–y plane. It is assumed that the steering angles at the front tires are equal, and thus $\delta^{FL} = \delta^{FR} = \delta$ holds. When driving into a turn, centrifugal forces act on the CG. The centrifugal force causes a cornering force and the corresponding side slip angles α_i (with $i \in \{FL, FR, RL, RR\}$). The calculation of the tire force is shown in the following section. The Newton equation for the vehicle chassis of the twin track model reads

$$m_v \ddot{\mathbf{r}}_v = \sum_{i \in \{FL,FR,RL,RR\}} \sum_{j \in \{k,s\}} \mathbf{F}_{j_i} + \mathbf{G} + \mathbf{F}_{\text{wind}}, \tag{15}$$

where $\mathbf{G}$ is the climbing resistance force and $\mathbf{F}_{\text{wind}}$ is the wind resistance force. The calculation of the horizontal tire forces $\mathbf{F}_{s_i}$ is presented in Sect. 3.1.2, while a detailed derivation of the vertical suspension forces $\mathbf{F}_{k_i}$ is out of scope of this paper. The Euler equation of the vehicle body reads

$$\Theta^v \cdot \dot{\boldsymbol{\omega}}^v + \boldsymbol{\omega}^v \times \left(\Theta \cdot \boldsymbol{\omega}^v\right) = \sum_{i \in \{FL,FR,RL,RR\}} \sum_{j \in \{k,s\}} \mathbf{r}_{j_i} \times \mathbf{F}_{j_i}, \tag{16}$$

where Θ^v is the inertia tensor of the vehicle body and $\boldsymbol{\omega}^v$ its angular velocity. The vectors $\mathbf{r}_{i_j}$ are defined analogously to Fig. 3.

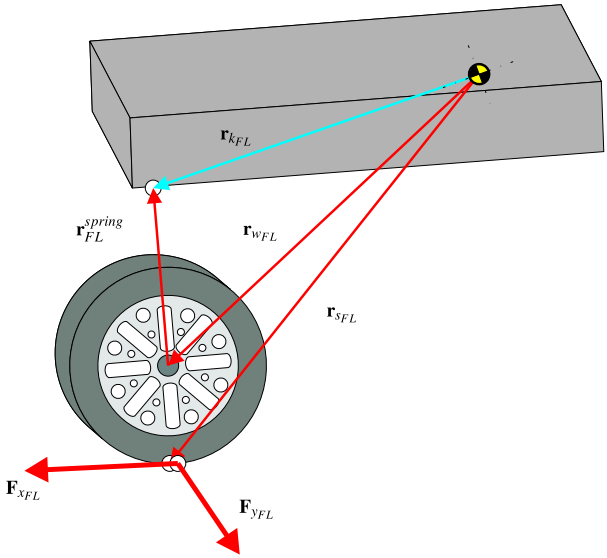


Fig. 3 Description of the front left quarter car suspension

3.1.2 Nonlinear tire model

The stationary tire forces are calculated through the Magic-Formula tire model. This tire model is based on Pacejka's Magic Formula [8] and belongs to the non-physical models. The stationary longitudinal and lateral tire forces are calculated depending on the longitudinal slip s_i, the slip angle α_i, the vertical wheel force $\mathbf{F}_{z_i}$ and the friction coefficient μ_i. The relation between the quantities is described by

$$\begin{bmatrix} F_{x_i}^{\text{stat}} \\ F_{y_i}^{\text{stat}} \end{bmatrix} = \begin{bmatrix} \frac{s_i}{s_{n,i}} \cdot F_i^{\text{total}} \\ \frac{\tan(\alpha_i)}{s_{n,i}} \cdot F_i^{\text{total}} \end{bmatrix}, \tag{17}$$

where

$$F_i^{\text{total}} = \sqrt{\frac{\tan^2(\alpha_i)}{s_{n,i}^2} \cdot F_{\text{ref},y_i}^2 + \frac{s_i^2}{s_{n,i}^2} F_{\text{ref},x_i}^2}, \tag{18}$$

$$F_{\text{ref},x_i} = \mu_x \sin(c_x \cdot \text{atan}(100 \cdot b_x s_{n,i})) \cdot F_{z_i}, \tag{19}$$

$$F_{\text{ref},y_i} = \mu_y \sin\left(c_y \cdot \text{atan}\left(\frac{180}{\pi} b_y \cdot \text{atan}(s_{n,i})\right)\right) \cdot F_{z_i}, \tag{20}$$

$$s_{n,i} = \sqrt{\tan^2(\alpha_i) + s_i^2} \tag{21}$$

for $i \in \{FL, FR, RL, RR\}$. Here $\mu_x, c_x, b_x, \mu_y, c_y$ and b_y are measured tire parameters. The dynamic wheel forces are calculated by means of a first-order differential equation which considers the running-in of the rubber particles of the tire:

$$\begin{bmatrix} \dot{F}_{x_i} \\ \dot{F}_{x_i} \end{bmatrix} = \begin{bmatrix} \frac{|r\omega_i^w|}{S_{x_i}} (F_{x_i}^{\text{stat}} - F_{x_i}) \\ \frac{|r\omega_i^w|}{S_{y_i}} (F_{y_i}^{\text{stat}} - F_{y_i}) \end{bmatrix}. \tag{22}$$

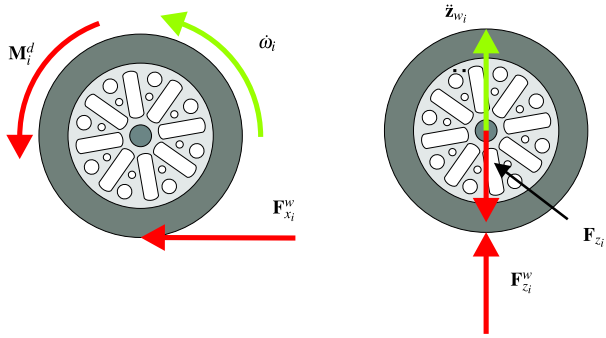


Fig. 4 Vertical and rotational movement of the wheel

Here r_i denotes the radius and ω_i^w the angular velocity of the ith wheel.

3.1.3 Equation of motion of the wheel

As stated before, each wheel possesses a translational d.o.f. which is the compression of the wheel suspension in the vertical z-direction (see Fig. 4). For this motion the Newton equation yields

$$m_w \ddot{\mathbf{z}}_{w_i} = \mathbf{F}_{z_i}^w - \mathbf{F}_{z_i} - m_w g \quad i \in \{FL, FR, RL, RR\}, \tag{23}$$

where $\mathbf{F}_{z_i}^w$ is the contact force, while $\mathbf{F}_{z_i}$ is the force acting from the suspension onto the wheel. Furthermore, the wheels have a rotational d.o.f. about the wheel axis (see Fig. 4). The rotational motion of the front and rear wheels is covered by the Euler Equation

$$\Theta^w \cdot \dot{\boldsymbol{\omega}}_i^w = \mathbf{M}_i^d - r_i \mathbf{F}_{x_i}^w \quad i \in \{FL, FR, RL, RR\}, \tag{24}$$

where $\mathbf{M}_i^d$ is the driving torque at the ith wheel, Θ^w denotes the inertia of the ith wheel and $\mathbf{F}_{x_i}^w$ is the longitudinal tire force.

3.2 Description of the single-track model

For the investigation of lateral dynamics while driving a vehicle into a turn, the two wheels at one axle can virtually be combined to one wheel element. This wheel element transmits the combined forces between the axle and the track. The single-track model gained in this approach is valid up to a lateral acceleration of about 4 m/s^2. This model covers three d.o.f. : the translational motion in x and y directions and the rotation about the yaw axis.

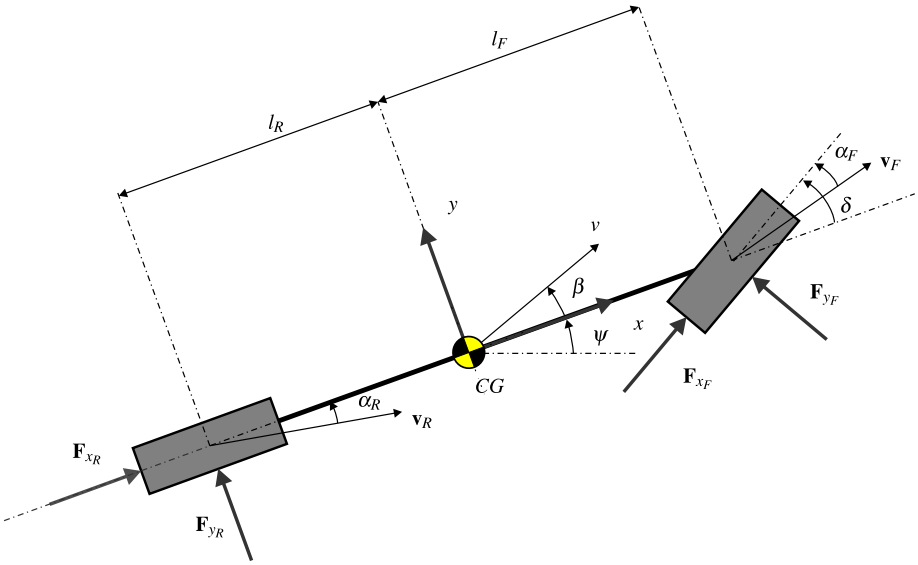


Fig. 5 Single track model

3.2.1 Equations of motion of the single-track model

All necessary quantities for describing the longitudinal and lateral dynamics of the single-track model are shown in Fig. 5. The equations of motion then read

$$\begin{bmatrix} \ddot{x}_{CG} \\ \ddot{y}_{CG} \\ \ddot{\psi} \\ \dot{\omega}_F \\ \dot{\omega}_R \\ \dot{F}_{x_F} \\ \dot{F}_{y_F} \\ \dot{F}_{x_R} \\ \dot{F}_{y_R} \end{bmatrix} = \begin{bmatrix} \frac{1}{m}(\mathrm{c}(\psi+\delta)F_{x_F} + \mathrm{c}(\psi)F_{x_R} - \mathrm{s}(\psi+\delta)F_{y_F} - \mathrm{s}(\psi)F_{y_R}) \\ \frac{1}{m}(\mathrm{s}(\psi+\delta)F_{x_F} + \mathrm{s}(\psi)F_{x_R} + \mathrm{c}(\psi+\delta)F_{y_F} + \mathrm{c}(\psi)F_{y_R}) \\ \frac{1}{\Theta^z}(l_f(\mathrm{s}(\delta)F_{x_F} + \mathrm{c}(\delta)F_{y_F}) - l_r F_{y_R}) \\ \frac{1}{\Theta^w}(M_A - r(\mathrm{c}(\delta)F_{x_F} - \mathrm{s}(\delta)F_{y_F})) \\ -\frac{1}{\Theta^w} r F_{x_R} \\ \frac{|r\omega_F|}{\sigma_x}(F_{x_F}^{\mathrm{stat}} - F_{x_F}) \\ \frac{|r\omega_F|}{\sigma_y}(F_{y_F}^{\mathrm{stat}} - F_{y_F}) \\ \frac{|r\omega_R|}{\sigma_x}(F_{x_R}^{\mathrm{stat}} - F_{x_R}) \\ \frac{|r\omega_R|}{\sigma_y}(F_{y_R}^{\mathrm{stat}} - F_{y_R}) \end{bmatrix}, \tag{25}$$

where the abbreviations

$$\mathrm{c} = \cos \quad \text{and} \quad \mathrm{s} = \sin$$

are used. The tire forces are calculated in the same manner as for the twin track model (see (17)–(22)), while the vertical forces and the slip are calculated by

$$F_{z_F} = \frac{l_r}{l} mg - \frac{l_z}{l}(\cos(\delta)F_{x_F} - \sin(\delta)F_{y_F}) + F_{x_R}, \tag{26}$$

$$F_{z_R} = \frac{l_f}{l} mg + \frac{l_z}{l}(F_{x_F} + F_{x_R}), \tag{27}$$

Table 1 Parameters of the single-track model

SYMBOL	VALUE	DESCRIPTION
l	2.7 m	Length of the vehicle
l_f	1.25 m	Distance from the front axle to CG
l_r	1.35 m	Distance from the front axle to CG
l_z	0.6 m	Distance from the ground to the CG
r	0.295 m	Radius of the tires
m	1200 kg	Mass of the vehicle
Θ^z	2400 kg·m^2	Vehicle moment of inertia
Θ^w	1.7 kg·m^2	Wheel moment of inertia
σ^x	0.01 m	Relaxation length
σ^y	0.2 m	Relaxation length
g	9.81 m/s^2	Gravity
μ_x, μ_y	0.9	Friction coefficients in the tire model
c_x, c_y, b_x, b_y	1.05, 0.3, 1.5, 0.15	Tire model parameters

$$v_{x_F} = \cos(\psi+\delta)(\dot{x}_{CG} - l_f\dot{\psi}\sin\psi) + \sin(\psi+\delta)(\dot{y}_{CG} + l_f\dot{\psi}\cos\psi), \tag{28}$$

$$v_{y_F} = -\sin(\psi+\delta)(\dot{x}_{CG} - l_f\dot{\psi}\sin\psi) + \cos(\psi+\delta)(\dot{y}_{CG} + l_f\dot{\psi}\cos\psi), \tag{29}$$

$$v_{x_R} = \cos(\psi)(\dot{x}_{CG} + l_r\dot{\psi}\sin\psi) + \sin(\psi)(\dot{y}_{CG} - l_r\dot{\psi}\cos\psi), \tag{30}$$

$$v_{y_R} = -\sin(\psi)(\dot{x}_{CG} + l_r\dot{\psi}\sin\psi) + \cos(\psi)(\dot{y}_{CG} - l_r\dot{\psi}\cos\psi), \tag{31}$$

$$s_i = \frac{v_{x_i} - r\omega_i}{\max(|r\omega_i|, |v_{x_i}|)}, \tag{32}$$

$$\alpha_i = -\arctan\left(\frac{v_{y_i}}{|r\omega_i|}\right), \tag{33}$$

$$s_{n,i} = \sqrt{\tan^2(\alpha_i) + s_i^2}, \tag{34}$$

for $i \in \{F, R\}$. The parameters of the single-track model are listed in Table 1.

4 Results

The reduction algorithm is implemented in MATLAB using the Maple Toolbox for MATLAB. Models can be imported using a MathML interface. After choosing the output variables, the error bound, the reduction technique(s) and the ranking procedure, the algorithm runs fully automated. Inside the algorithm the equations are treated as tree-like structures in order to manipulate any term or subterm. The residual functions for the simulations are generated as C-code using the Maple code export and compiled as MATLAB mex-files to accelerate the evaluation. The emerging DAE-systems are solved either using IDAS of the SUNDIALS package which is a C implementation of DASPK [2, 6] and offers a MATLAB interface or the MATLAB built-in solver ode15i. The time that is needed depends highly on the chosen model, the ranking procedure and the reduction technique. Here, the reduction technique has a big influence, since linearization and setting to constants needs a newly compiled residual function for every simulation, while cancellation of terms works with only

one simulation file. Moreover, the generation and compilation of the residual function are much slower compared to commercial tools like Dymola. All calculations presented here are performed on a 1.8 GHz laptop.

4.1 Reduction of the single-track model

In this section, the single-track model presented in Sect. 3.2 is reduced in order to explain the reduction algorithm clearly. Since one cannot expect many reductions choosing cancellation of terms as reduction technique, the linearization of terms is chosen. The scenario is a combination of an accelerated straight-ahead travel and a double lane change. More precisely, the vehicle is accelerated in the first 8 seconds from 8 to 17.5 m/s followed by a double lane change lasting 20 seconds. The output variables are chosen to be the longitudinal velocity $\dot{x}_{CG}$, the lateral velocity $\dot{y}_{CG}$ and the yaw velocity $\dot{\psi}_{CG}$. In the following, the single-track model is first reduced for two different ranking procedures using the same error bound. After that, one of the already reduced models is further reduced raising the error bound. In order to compare the efficiency of the models, the number of floating point operations (FLOPs) for one integration step is determined. As integration algorithm the semi-implicit Euler method is chosen, since it is widely used for real-time simulations. One integration step with the semi-implicit Euler method requires one evaluation of the right-hand side, one evaluation of the Jacobian and the solution of a system of linear equations. This gives 34551 FLOPs for the original single-track model.

In a first step, the error bound is set to 1.5%. Using the Residual Ranking leads to

$$\begin{bmatrix} \ddot{x}_{CG} \\ \ddot{y}_{CG} \\ \ddot{\psi} \\ \dot{\omega}_F \\ \dot{\omega}_R \\ \dot{F}_{x_F} \\ \dot{F}_{y_F} \\ \dot{F}_{x_R} \\ \dot{F}_{y_R} \end{bmatrix} = \begin{bmatrix} \frac{1}{m}(F_{x_F} + F_{x_R} - (\psi + \delta)F_{y_F} - \psi \cdot F_{y_R}) \\ \frac{1}{m}((\psi + \delta)F_{x_F} + \psi \cdot F_{x_R} + F_{y_F} + F_{y_R}) \\ \frac{1}{\Theta^z}(l_f(\delta \cdot F_{x_F} + F_{y_F}) - l_r F_{y_R}) \\ \frac{1}{\Theta^w}(M_A - r(F_{x_F} - \delta \cdot F_{y_F})) \\ -\frac{1}{\Theta^w} r F_{x_R} \\ \frac{|r\omega_F|}{\sigma_x}(F_{x_F}^{\text{stat}} - F_{x_F}) \\ \frac{|r\omega_F|}{\sigma_y}(F_{y_F}^{\text{stat}} - F_{y_F}) \\ \frac{|r\omega_R|}{\sigma_x}(F_{x_R}^{\text{stat}} - F_{x_R}) \\ \frac{|r\omega_R|}{\sigma_y}(F_{y_R}^{\text{stat}} - F_{y_R}) \end{bmatrix}. \tag{35}$$

Moreover, the calculation of the steady-state tire forces reduces to

$$\begin{bmatrix} F_{x_i}^{\text{stat}} \\ F_{y_i}^{\text{stat}} \end{bmatrix} = \begin{bmatrix} \frac{s_i}{s_{n,i}} \cdot F_i^{\text{total}} \\ \frac{\alpha_i}{s_{n,i}} \cdot F_i^{\text{total}} \end{bmatrix}, \tag{36}$$

where

$$F_F^{\text{total}} = \sqrt{\frac{\tan(\alpha_F) \cdot \alpha_F}{s_{n,F}^2} \cdot F_{\text{ref},y_F}^2 + \frac{s_F^2}{s_{n,F}^2} F_{\text{ref},x_F}^2}, \tag{37}$$

$$F_R^{\text{total}} = \sqrt{\frac{\alpha_R^2}{s_{n,R}^2} \cdot F_{\text{ref},y_R}^2 + \frac{s_R^2}{s_{n,R}^2} F_{\text{ref},x_R}^2}, \tag{38}$$

$$F_{\text{ref},x_F} = \mu_x \sin(100 \cdot c_x b_x s_{n,F}) \cdot F_{z_F}, \tag{39}$$

$$F_{\mathrm{ref},x_R} = 100 \cdot \mu_x c_x b_x s_{n,R} \cdot F_{z_R}, \tag{40}$$

$$F_{\mathrm{ref},y_F} = \mu_y c_y \cdot \mathrm{atan}\left(\frac{180}{\pi} b_y \cdot s_{n,F}\right) \cdot F_{z_F}, \tag{41}$$

$$F_{\mathrm{ref},y_R} = \mu_y \sin\left(c_y \cdot \mathrm{atan}\left(\frac{180}{\pi} b_y \cdot s_{n,R}\right)\right) \cdot F_{z_R}, \tag{42}$$

$$\alpha_i = -\frac{v_{y_i}}{|r\omega_i|}, \tag{43}$$

$$s_{n,i} = \sqrt{\alpha_i^2 + s_i^2}, \tag{44}$$

for $i \in \{F, R\}$. The calculation of the vertical forces, the velocities and the longitudinal slip remains the same as in (26)–(33). It can be seen that the steering and yaw angle are linearized in the equations of motion, but not for the calculation of the velocities at the front and the rear. Additionally, the tire model is already simplified. This reduced model requires only 19116 FLOPs.

In the next step the same reduction is performed once again, but this time using the One-Step Ranking. In this case the model is the same as the previously reduced model except for some equations. The second line of (35) gets

$$\ddot{y}_{CG} == \frac{1}{m}(\sin(\psi + \delta) F_{x_F} + \psi \cdot F_{x_R} + F_{y_F} + F_{y_R}), \tag{45}$$

while (37), (39) and (42) become

$$F_F^{\mathrm{total}} = \sqrt{\frac{\alpha_F^2}{s_{n,F}^2} \cdot F_{\mathrm{ref},y_F}^2 + \frac{s_F^2}{s_{n,F}^2} F_{\mathrm{ref},x_F}^2}, \tag{46}$$

$$F_{\mathrm{ref},x_F} = 100 \cdot \mu_x c_x b_x s_{n,F} \cdot F_{z_F}, \tag{47}$$

$$F_{\mathrm{ref},y_R} = \mu_y c_y \cdot \mathrm{atan}\left(\frac{180}{\pi} b_y \cdot s_{n,R}\right) \cdot F_{z_R}. \tag{48}$$

Furthermore, this time (28) and (30) are reduced to

$$v_{x_F} = \cos(\psi + \delta)(\dot{x}_{CG} - l_f \dot{\psi} \sin\psi) + (\psi + \delta)(\dot{y}_{CG} + l_f \dot{\psi} \cos\psi), \tag{49}$$

$$v_{x_R} = \cos(\psi)(\dot{x}_{CG} + l_r \dot{\psi} \sin\psi) + \psi \cdot (\dot{y}_{CG} - l_r \dot{\psi} \cos\psi). \tag{50}$$

This new model requires only 14746 FLOPs for one integration step using the semi-implicit Euler method. Hence, the One-Step Ranking was more efficient in this case. Although this result cannot be generalized, in most cases the One-Step Ranking is more efficient due to the more accurate ranking list. However, the big drawback of the One-Step Ranking can be seen when looking at the run-times of the algorithms. While the whole reduction using the Residual Ranking takes about 15 minutes, the reduction algorithm using the One-Step Ranking needs about 50 minutes.

In a last step the error bound is raised to 5%. Using the Residual Ranking, the equations of motion are the same as in (35). The calculation of the steady-state tire forces is performed in the same way as for the reduced model using the One-Step Ranking. However, the calculation of the velocities at the front and the rear reduces to

$$v_{x_F} = (\dot{x}_{CG} - l_f \dot{\psi} \sin\psi) + (\psi + \delta)(\dot{y}_{CG} + l_f \dot{\psi} \cos\psi), \tag{51}$$

$$v_{y_F} = -(\psi + \delta)(\dot{x}_{CG} - l_f \dot{\psi} \sin \psi) + \cos(\psi + \delta)(\dot{y}_{CG} + l_f \dot{\psi} \cos \psi), \tag{52}$$

$$v_{x_R} = (\dot{x}_{CG} + l_r \dot{\psi} \sin \psi) + \psi \cdot (\dot{y}_{CG} - l_r \dot{\psi} \cos \psi), \tag{53}$$

$$v_{y_R} = -\psi \cdot (\dot{x}_{CG} + l_r \dot{\psi} \sin \psi) + \cos(\psi)(\dot{y}_{CG} - l_r \dot{\psi} \cos \psi). \tag{54}$$

Note that this model is not the linear single-track model, although it is similar. It requires only 11908 FLOPs for one integration step, which is approximately one third of the required FLOPs for the original model. Obviously, the quality of the reduced models depends on the error bound, the reduction technique and the ranking.

4.2 Reduction of the twin track model

In this section, the spatial twin track model presented in Sect. 3.1 is selected as the starting model. The parameters of the model are chosen in order to match a medium-class vehicle. The reduction is performed for different scenarios, covering a cornering maneuver, an accelerated cornering maneuver and a double lane change. Negligence of terms and symmetry considerations are used as reduction techniques, while the One-Step Ranking as well as the Residual Ranking are applied for the computation of the ranking list. The yaw rate and the lateral acceleration are chosen as output variables since they represent the significant dynamic behavior. In the next sections the reduced models are presented. The speed-up is approximately a factor of six for all three scenarios.

4.2.1 Accelerated cornering maneuver

As a first attempt, the spatial twin track model is reduced for an accelerated cornering maneuver. The steering angle at the front wheels is continuously raised from 0° to 1° within the first second of the maneuver. The whole maneuver lasts 10 seconds. Starting from an initial velocity of 12 m/s, the vehicle is accelerated with a driving torque of 490 N m. The reduction algorithm started using the negligence of terms as reduction technique with an error bound of 10% and finished with symmetry considerations using an error bound of 15%. Moreover, the One-Step Ranking was employed as ranking procedure. After the negligence of terms, the vertical movement of the wheels is vanished as well as *yaw*- and roll movement. Therefore, at this step a planar twin track model is at hand. In the second step, it is checked whether similar variables can be replaced by only one variable. In this example, the pairs of variables to be checked were marked for the algorithm. These symmetry considerations end up with a single-track model. Moreover, the number of d.o.f. is reduced to five. The tire forces are now calculated using a simplified Magic Formula, since some subterms are neglected. The resulting yaw rate and lateral acceleration are plotted in Figs. 6 and 7. Figure 6 indicates that the error becomes smaller as the simulation time rises. This effect is due to two reductions which lead to errors in opposite directions. In the beginning the influence of the first reduction is superior to the influence of the second reduction and thus the lateral acceleration of the reduced model is lower than the lateral acceleration of the original model. After some time the influence of the second reduction grows and the error becomes smaller. Consequently, for simulation times longer than 10 seconds, the lateral acceleration of the reduced model would be higher than the lateral acceleration of the spatial twin track model. The same effect can be seen in Fig. 7. However, here the influence of the second reduction gets superior after approximately six seconds. The original and the reduced model are shortly summarized in Table 2.

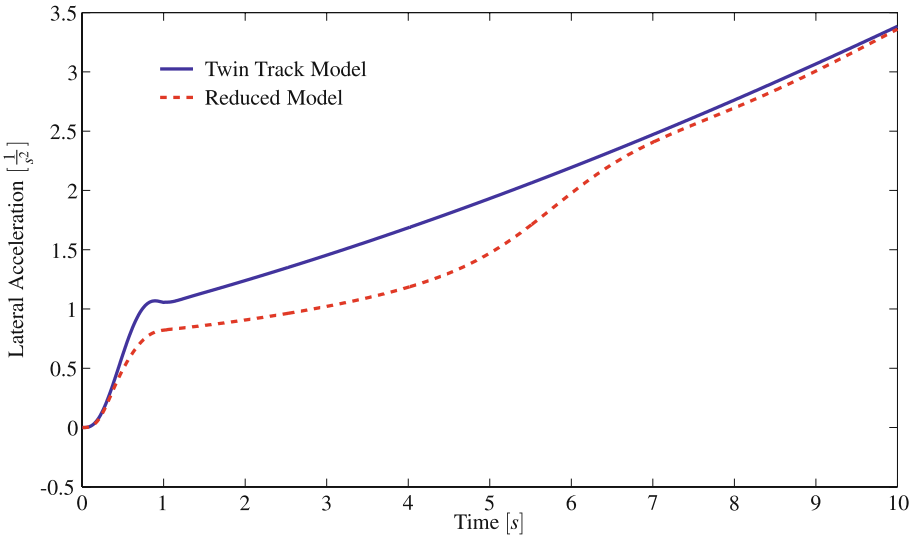


Fig. 6 Lateral acceleration of the twin track model and the reduced model for the accelerated cornering maneuver

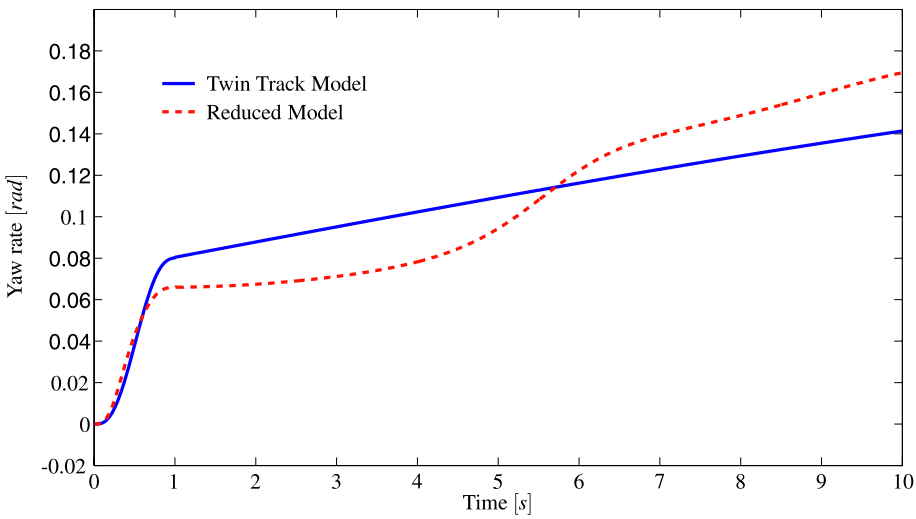
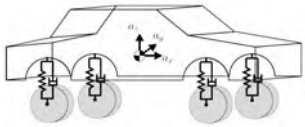
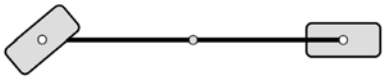


Fig. 7 Yaw rate of the twin track model and the reduced model for the accelerated cornering maneuver

4.2.2 Cornering maneuver

The second scenario is very similar to the first one. The only difference is in the driving torque. While the cornering maneuver of the previous section was performed with a driving torque of 490 N m, this time the driving torque is set to zero during the whole simulation. Also, the reduction algorithm stays nearly unchanged. Only the One-Step Ranking was exchanged by the Residual Ranking. The Residual Ranking leads to a reduction result which

Table 2 Comparison of the original and the reduced model for the accelerated cornering maneuver

VEHICLE MODEL	DEGREES OF FREEDOM	SCHEMATIC REPRESENTATION
Spatial Twin Track Model	d.o.f.: 14 – Three Translational – Roll, Pitch, Yaw – Rotation of the wheels – vertical movement of the wheels	
Reduced Model	d.o.f.: 5 – Two Translational – Yaw – Rotation of the wheels	

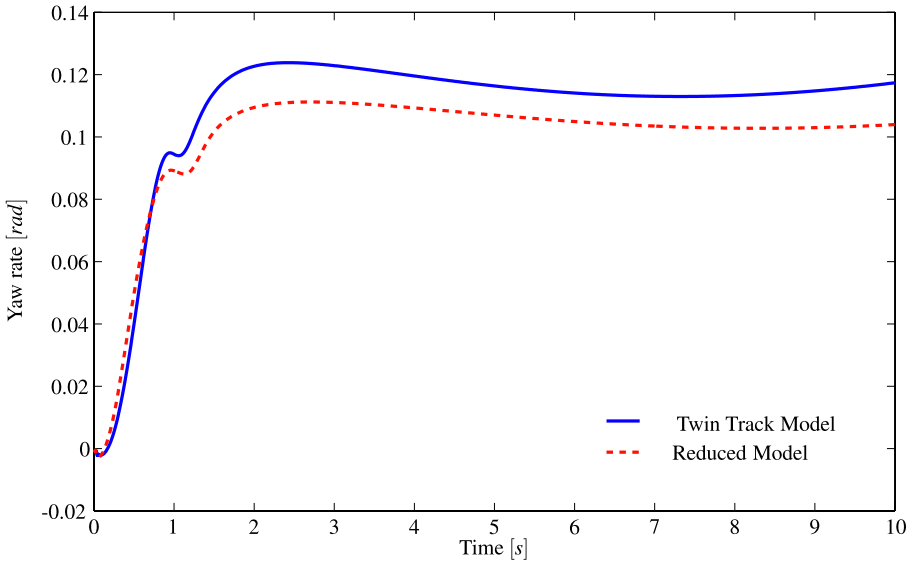


Fig. 8 Yaw rate of the twin track model and the reduced model for the cornering maneuver

differs from the result of the previous section as can be seen in Figs. 8 and 9. That may be astonishing at first sight. However, the Residual Ranking may produce a less reasonable ranking list (reductions sorted by their ranking values). If the error estimated by Residual Ranking is small, then the real error will be small as well. The opposite is not true in general. More precisely, a reduction might cause a high ranking value, although the real error emerging from that reduction is quite small [14]. Consequently, the ranking list for the Residual Ranking may be completely different from the ranking list of the One-Step Ranking. Hence, the reductions are checked in a different order and thus different reductions are performed. Here the error bound was again set to real values representing approximately 15%. As in the previous example, the vertical motion of the wheels vanishes. Nevertheless, the angular

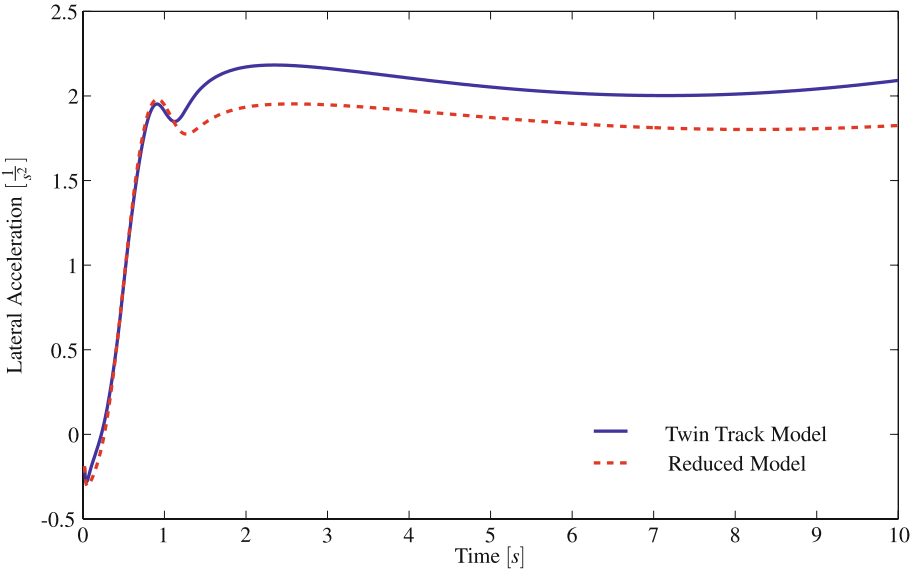


Fig. 9 Lateral acceleration of the twin track model and the reduced model for the cornering maneuver

velocity of the rotations about the x and y axes remains, while the angles get decoupled from their derivatives and stay constant. Moreover, this time the result is not a single-track model. Using the Residual Ranking, the negligence of terms leads to a model which has a quite asymmetric shape. Hence, symmetry considerations lead to a union of the tire forces in x-direction and to the union of the rotational motion of the front wheels. The tire forces in y direction and the rotational motion of the rear wheels are still calculated separately. Therefore, the reduced model has 8 d.o.f., since this time the rotation of three wheels remains as well as pitch and roll motion (in some sense). A scheme of the reduced model in the x–y plane is shown in Fig. 10. In order to obtain results which are easier to interpret, one should either use the One-Step Ranking or perform the symmetry reduction before the negligence of terms. In this contribution the order of the reduction techniques was chosen the other way round for performance reasons.

4.2.3 Double lane change

The third scenario examined here is a double lane change at a velocity of 12 m/s. For performance reasons, again the Residual Ranking was employed. Furthermore, an error bound of approximately 20% is used. In contrast to the second example, here the Residual Ranking gives good estimates of the real error produced by the reductions. The negligence once again cancels out the vertical motion of the four wheels. This time, additionally roll and pitch motion vanishes. Moreover, the model maintains its symmetry during the first step of the reduction. Hence, the symmetry reduction merges the tire forces in x and y directions as well as the rotational motion of the front and rear tires. Therefore, the result of the reduction is a vehicle model with only one track and five d.o.f. Thus, it is similar to the result of the first example. However, here more terms of the tire model are preserved. The resulting yaw rate and lateral acceleration are presented in Figs. 11 and 12, respectively.

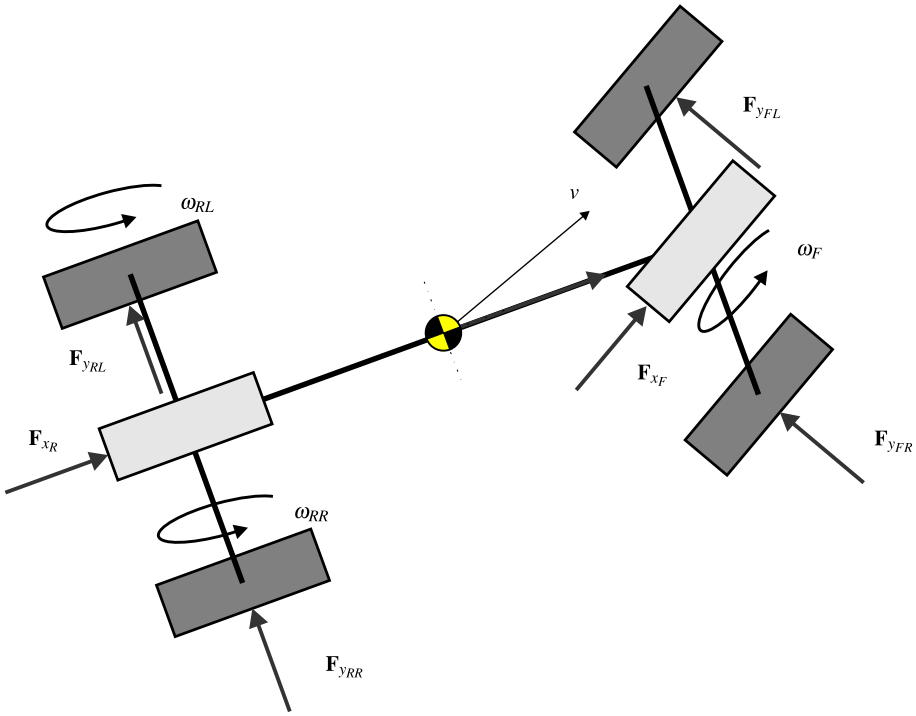


Fig. 10 Reduced model in the $x-y$ plane

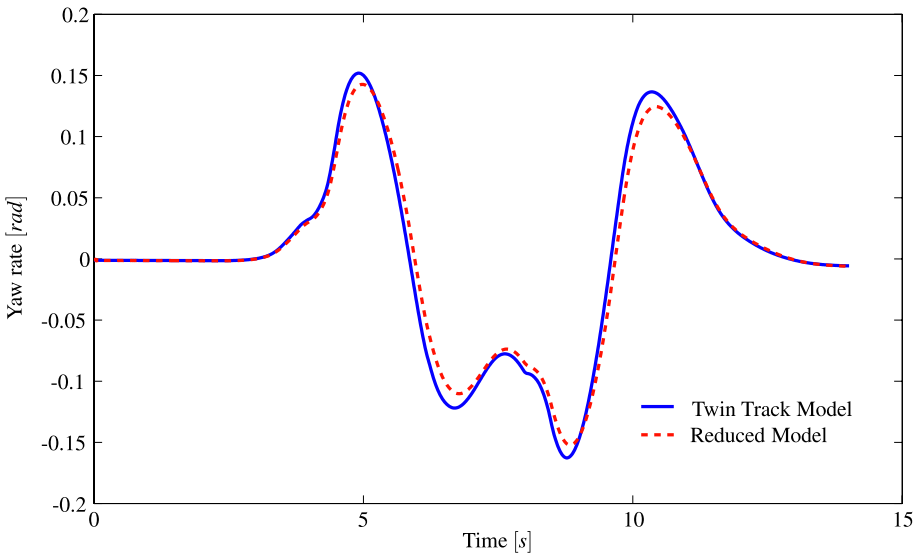


Fig. 11 Yaw rate of the twin track model and the reduced model for the double lane change

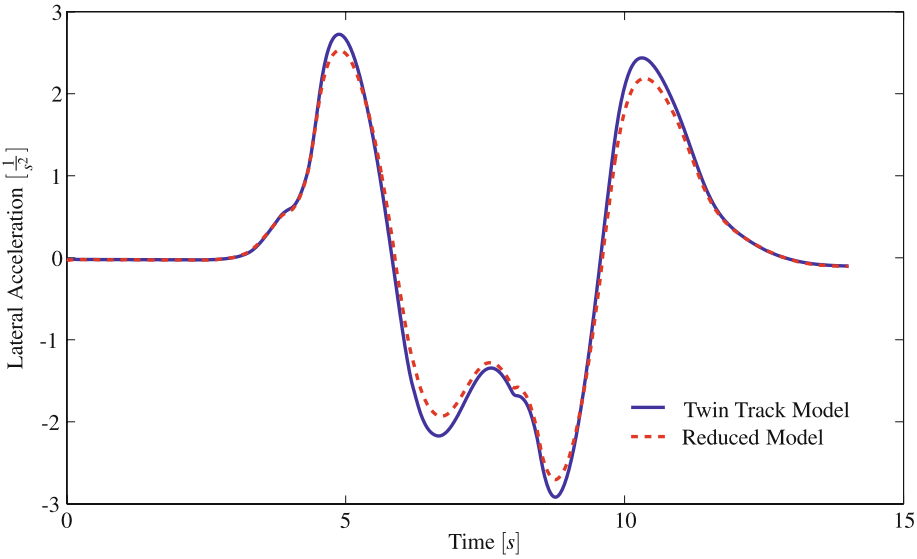


Fig. 12 Lateral acceleration of the twin track model and the reduced model for the double lane change

5 Conclusion and outlook

In this contribution, a new systematic approach for the generation of vehicle models is proposed. Moreover, four examples are given. The results obtained cover conventional models (Sects. 4.2.1 and 4.2.2) as well as non-conventional vehicle models (Sect. 4.2.3). It was seen that while the Residual Ranking is beneficial in computation time, the One-Step Ranking delivers more accurate results. In the near future the algorithm will be implemented as a plug-in for the Open Modelica compiler [4] in order to reduce run-time. Moreover, the algorithm will be extended in order to generate models that can be simulated in real time on a predefined real-time target. Hence, not an error bound is given, but the real-time hardware and the real-time cycle. First, results are presented in [7]. Furthermore, a new ranking procedure based on the adjoint sensitivity analysis will be implemented. Last but not least, it may be of interest to protect some terms from being removed by the reduction algorithm. Having a vehicle model coupled with an ESP at hand, the question for the required complexity of the vehicle model to examine the ESP arises. Hence, the ESP should be protected from being simplified. That feature (Modular Reduction) will also be integrated into the algorithm in the future.

References

1. Borchers, C.: Symbolic behavioral model generation of nonlinear analog circuits. IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process. **45**(10), 1362–1371 (1998)
2. Brenan, K.E., Campbell, S.L.V., Petzold, L.R.: Numerical solution of Initial-Value Problems in Differential-Algebraic Equations. Society for Industrial Mathematics, Philadelphia (1996)
3. Burgermeister, B., Arnold, A., Eichberger, A.: Smooth velocity approximation for constrained systems in real-time simulation. In: Proceedings of the ECCOMAS Thematic Conference in Multibody Dynamics, Warsaw, Poland (2009)

4. Fritzson, P., Aronsson, P., Bunus, P., Engelson, V., Saldamli, L., Johansson, H.: The open source Modelica project. In: Proceedings of the 2nd International Modelica Conference, Oberpfaffenhofen, Germany (2002)
5. Gorban, A.N., Kazantzis, N., Kevrekidis, I.G., Ottinger, H.C., Theodoropoulos, C.: Model Reduction and Coarse-Graining Approaches for Multiscale Phenomena. Springer, Berlin, Heidelberg (2006)
6. Hindmarsh, A.C., Brown, P.N., Grant, K.E., Lee, S.L., Serban, R., Shumaker, D.E., Woodward, C.S.: SUNDIALS: suite of nonlinear and differential-algebraic equation solvers. ACM Trans. Math. Softw. **31**(3), 363–396 (2005)
7. Mikelsons, L., Ji, H., Brandt, T., Lenord, O.: Symbolic model reduction applied to real-time simulation of a construction machine. In: Proceedings of the 7th International Modelica Conference, Como, Italy (2009)
8. Pacejka, H.B.: Tyre and Vehicle Dynamics. Butterworth-Heinemann, Oxford (2006)
9. Popp, K., Schiehlen, W.: Ground Vehicle Dynamics. Springer, Berlin, Heidelberg (2010)
10. Riekert, P., Schunck, T.: Zur Fahrmechanik des gummibereiften Kraftfahrzeugs. Arch. Appl. Mech. **11**(3), 210–224 (1940)
11. Sommer, R., Halfmann, T., Broz, J.: Automated behavioral modeling and analytical model-order reduction by application of symbolic circuit analysis for multi-physical systems. Simul. Model. Pract. Theory **16**(8), 1024–1039 (2008)
12. Wichmann, T.: Computer aided generation of approximate DAE systems for symbolic analog circuit design. J. Appl. Math. Mech. **81**, 633–644 (2000)
13. Wichmann, T.: Transient ranking methods for the simplification of nonlinear DAE systems in analog circuit design. Proc. Appl. Math. Mech. **2**(1), 448–449 (2003)
14. Wichmnann, T.: Symbolische Reduktionsverfahren für nichtlineare DAE-Systeme. Ph.D. thesis, Technische Universität Kaiserslautern, Germany (2004)