Applications

Stefan Barthelmes* and Ulrich Konigorski

Model-based chassis control system for an over-actuated planetary exploration rover

Modellbasierte Chassisregelung für einen überaktuierten planetaren Explorationsrover

https://doi.org/10.1515/auto-2019-0090
Received July 31, 2019; accepted November 12, 2019

Abstract: In planetary exploration, wheeled mobile robots (rovers) are popular for extending action range compared to a lander. Despite their success, they continue to struggle with soft grounds which shows in high sinkage and can lead to an immobilization in the worst case. Rovers usually are over-actuated due to individual wheel drives and steering, which is rarely made use of in current missions. Some work optimizing the resulting degrees of freedom exists but often does not use all available model knowledge. In this work, the rover is consequently modeled with the subsystems rigid body dynamics, kinematics and wheel/ground dynamics. Feedback linearization is used for the rigid body and the underlying wheel/ground controllers on individual wheel level. The control allocation of the forces is done via the pseudo-inverse and a base of the null-space to extract the available degrees of freedom. A verification of the approach is shown in a co-simulation with a high-fidelity model of the ExoMars rover.

Keywords: chassis control, mobile robots, planetary exploration rover, torque control, traction optimization, control allocation, feedback linearization, ExoMars, dynamic extension

Zusammenfassung: Radgetriebene, mobile Roboter - sogenannte Rover - sind in der planetaren Exploration ein verbreitetes Mittel zur Erhöhung des Aktionsradiuses im Vergleich zu stationären Landesonden. Obwohl Rover in vergangenen Missionen sehr erfolgreich waren, haben sie nach wie vor Probleme mit dem Vorankommen auf Sandböden, dies spiegelt sich in hoher Einsinkung wider und kann dazu führen, dass sich der Roboter festfährt. Aufgrund einzeln angetriebener und gelenkter Räder sind Rover häufig überaktuiert, allerdings wird von den dadurch entstehenden Möglichkeiten in Missionen kaum Gebrauch gemacht. Während in der Forschung einige Arbeiten zur Optimierung der entstehenden Freiheitsgrade existieren, wird dabei oft nur partiell Modellwissen verwendet. In dieser Arbeit wird der Rover vollständig mit seinen Subsystemen Starrkörperdynamik, Kinematik und Rad-/Bodenkontaktdynamik modelliert. Die Modelle werden anschließend modifiziert, sodass für die dynamischen Subsysteme Regelgesetze mit der Methode der exakten Ein-/Ausgangslinearisierung entworfen werden können, während die Allokation zu einem statischen Problem wird. Diese Kraftverteilung wird mithilfe der Pseudo-Inversen und einer Basis des Nullraums der Allokationsmatrix realisiert. Der Ansatz wird in einer Co-Simulation mit einem genauen Modell des ExoMars Rovers verifiziert.

Schlagwörter: Chassisregelung, Mobile Roboter, Planetare Explorationsrover, Drehmomentregelung, Traktionsoptimierung, Allokation, Exakte Ein-/Ausgangslinearisierung, dynamische Erweiterung, ExoMars

1 Introduction

Wheeled mobile robots significantly increase the action range for scientific payloads compared to immobile landers. Due to the importance to scientists and the fact, that rovers allow exploration and surveillance of hazardous areas without threatening astronauts [10, p. 4ff], rovers have become popular in planetary exploration. Most prominently, the NASA/JPL rovers Spirit, Opportunity and Curiosity have been and are enormously successful in exploring the Mars in the past 1.5 decades. Despite this success, the mobility of these rovers in heterogeneous and especially soft and sandy terrain continues to pose problems. The Spirit rover, for example, got stuck in 2009 due to excessive sinkage of one wheel and Curiosity was backed out of a sand field in Hidden Valley in 2014 due to dangerously

***Corresponding author: Stefan Barthelmes,** Deutsches Zentrum für Luft- und Raumfahrt e.V., Institut für Systemdynamik und Regelungstechnik, Münchener Str. 20, 82234 Weßling, Germany, e-mail: stefan.barthelmes@dlr.de
Ulrich Konigorski, TU Darmstadt, Institut für Automatisierungstechnik und Mechatronik, Fachgebiet Regelungstechnik und Mechatronik, Landgraf-Georg-Str. 4, 64283 Darmstadt, Germany, e-mail: ukonigorski@iat.tu-darmstadt.de

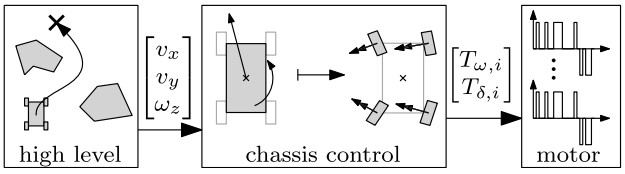

Figure 1: Overall controls hierarchy for a wheeled mobile robot; the mid-layer chassis control is the subject of this work.

increasing sinkage [2, 3]. Because of scientific interest and the mechanical threat of driving over spiky rocks instead of sand, it is worth improving the driving capabilities in soft sands. A model-based controller is designed, in this work, to increase rover performance by software without changing the rover design.

For a mobile robot, the control task can be divided into three main layers, see figure 1. The first layer ('high level') is responsible for creating a desired robot velocity trajectory and can include tasks such as navigation, obstacle avoidance, path optimization and path following. Motor controllers, on the other hand, usually accept current, velocity or position trajectories and control the motors accordingly. This work deals with the chassis control, which is the control layer in between the high level and the motor controllers. The chassis controller is hence in charge of computing desired values for each actuator from the desired robot velocities. There are more controllable degrees of freedom than would be necessary due to mobile robots mostly having individual wheel drive and sometimes steering to achieve a planar movement. *Control allocation* is the controls domain that deals with the resulting over-actuated system or, mathematically speaking, underdetermined system of equations; see the survey paper of Johansen and Fossen [20] for a general overview.

Within the area of wheeled mobile robots, existing approaches can be divided into (wheel) rate-based and (wheel) torque-based controllers. Most often, a rate-based approach, with simple geometric constraints to handle the underdeterminacy of the system, is used as described in [14, 25]. This control approach is often called *geometric control* and can be thought of as an extension of the Ackermann steering angles for vehicles with two-wheel steering. While slip is mostly assumed to be zero, Ishigami et al. [18] add slippage considerations to the wheel rate calculations and Helmick et al. [14] account for slip in the higher level path following. Another approach for very rough terrain is to account for the directions of movement of the individual wheels and adjust the wheel rates such that slip is minimized by preventing the so-called *wheel-fighting*. This technique relies on an estimation or measurement of the individual wheel contact angles (see [16]), was developed by Peynot and Lacroix [32] and recently updated for and uploaded to JPL's Curiosity rover [36]. A switching controller, combining standard geometric control, slippage consideration and contact angles, was proposed by Gonzalez and Iagnemma [11]. Krenn et al. [24] presented a model-predictive control (MPC) which considers terramechanics properties within a rate-based approach.

The first torque controllers were built in the field of mobile robots with non-holonomic constraints that assume zero slippage. There are approaches to cope with steered wheels [35], a four-wheeled skid-steer rover with lateral but no longitudinal wheel slip [8, 22] and a two-wheeled differential drive robot that is subjected to longitudinal wheel slip [39]. Other approaches explicitly control a desired slip value [27], that is calculated through an inversion of simple terramechanics models, or switch between dedicated geometrically derived control strategies for different terrains [37]. Most torque-based controllers are, however, based on allocating the desired wheel torques with an inversion of the robot kinematics. To cope with the over-actuation, either the least squares solution [27] is used or an optimization is employed to find the best distribution. A minimization of the power consumption [17, 34], the ratio between traction force and the normal force [26, 16] or a stability measure to prevent tip-over of the robot [28] was performed. While these works either use PID-controllers to calculate the total forces needed or do not mention that part of the controller at all, there are approaches that use models to compute the wheel forces needed for maintaining a static force-torque balance. Movement of the robot is then enabled by an additional PID controller on the velocity error. These approaches were published by Michaud et al. [29] for the ExoMars rover of the European Space Agency (ESA) and for a research rover by Krebs et al. [23].

The field of research of road vehicles equipped with individually driven and steered wheels has relevance in this project as well because these vehicles have similarities with mobile robots, cf. Krenn et al. [25] for a comparison. Excessive work was published [31, 15, 30, 6] and the same controls hierarchy as depicted in figure 1, can be observed.

In the present work, a comprising model-based chassis control system for the ESA ExoMars rover [29] is developed to control all six drive and six steering actuators, such that the rover follows a desired velocity trajectory. Therefore, synthesis models for a 3D rigid body dynamics and full kinematics of the passive suspension system,

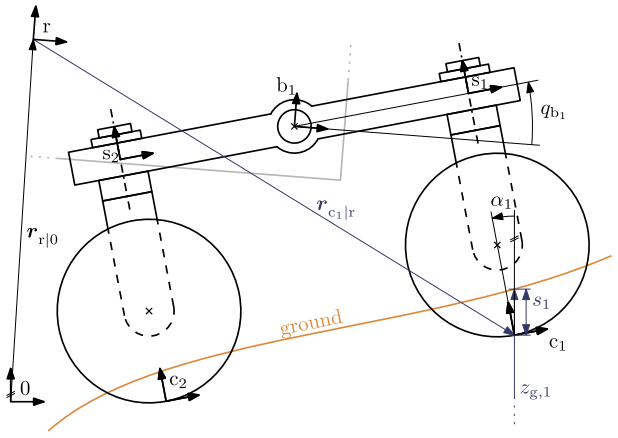

Figure 2: One bogie with its two wheel units and quantities for the sinkage calculation for a penetration based normal force model.

as well as for the wheel and ground dynamics are developed. Three controller modules are designed, see figure 3: The 3D rigid body dynamics model is modified such that it can be controlled with a feedback linearization controller. The control allocation is done with the pseudo-inverse of the partial force-torque balance equation, while the available degrees of freedom are found with the null-space and optimized in a separate lower frequency module. The wheel-ground dynamics model is dynamically extended to be used within another feedback linearization for the synthesis of the underlying single wheel controllers.

For this approach, one contact point with the ground per wheel and a contact angle (see [16]) of zero is assumed, which corresponds to the assumption of moderate terrain roughness. Furthermore, the rover translational and rotational velocities, as well as the orientation, bogie angles, wheel rates and steering angles, are assumed to be ideally measurable.

2 System model

The coordinate systems that are used for the model equations are defined according to the kinematics of the ExoMars rover. The passive suspension system of the rover consists of three so-called *bogies*, which are crossbeams that hold two wheel assemblies each and are mounted with a passive rotational degree of freedom to the rover body. One of the bogies is depicted in figure 2, where the inertial 0-frame, the rover center of mass (CoM) frame ('r'), the bogie frame ('b'), steering frame ('s') and contact frame ('c') are shown. A right lower index ${}^{\mathrm{r}}(\cdot)_{\mathrm{c}_i|\mathrm{r}}$, in this work, means *the ith contact frame with respect to the* r*-frame*, while the upper left index indicates the frame in which the vector is resolved. For shorter notation, the upper left index is omitted, if the describing frame is equal to the originating one, $\boldsymbol{r}_{\mathrm{c}_i|\mathrm{r}} = {}^{\mathrm{r}}\boldsymbol{r}_{\mathrm{c}_i|\mathrm{r}}$.

The ExoMars rover, including its ground-interaction, can be divided into three major components. Robot dynamics are considered as 3-dimensional rigid body dynamics to calculate the movements that are achieved by total forces and torques, see section 2.1. For the static equation system to map these total forces and torques to the wheel forces, the force-torque balance is derived through the rover kinematics and differential kinematics in section 2.2.

On single-wheel level, the steering and drive dynamics, as well as a simple wheel-ground contact model, are combined to one dynamic system in section 2.3.

2.1 Rigid body dynamics

A 3D dynamic model is developed due to the rover being able to move on uneven terrain, including slopes of more than 15°. The dynamics of the bogies are neglected and one lumped mass matrix $\underline{\boldsymbol{M}}_{\mathrm{r}}$ and one inertia matrix $\underline{\boldsymbol{I}}_{\mathrm{r}}$ are assumed for the whole rover, including its suspension and wheels. The equations of motion can then be written as the nonlinear state-space system

$$\underbrace{\begin{bmatrix}\dot{\boldsymbol{v}}_{\mathrm{r}}\\ \dot{\boldsymbol{\omega}}_{\mathrm{r}}\\ \dot{\boldsymbol{\theta}}\end{bmatrix}}_{\dot{\boldsymbol{x}}_{\mathrm{r}}} = \underbrace{\begin{bmatrix}-\underline{\boldsymbol{S}}(\boldsymbol{\omega}_{\mathrm{r}})\boldsymbol{v}_{\mathrm{r}} + {}^{\mathrm{r}}\underline{\boldsymbol{R}}_0(\boldsymbol{\theta})\,{}^0\boldsymbol{g}\\ \underline{\boldsymbol{0}}\\ \underline{\boldsymbol{U}}^{-1}(\boldsymbol{\theta})\boldsymbol{\omega}_{\mathrm{r}}\end{bmatrix}}_{\boldsymbol{f}_{\mathrm{r}}(\boldsymbol{x}_{\mathrm{r}})} + \underbrace{\begin{bmatrix}\frac{1}{\underline{M}_{\mathrm{r}}} & \underline{\boldsymbol{0}}\\ \underline{\boldsymbol{0}} & \frac{1}{\underline{I}_{\mathrm{r}}}\\ \underline{\boldsymbol{0}} & \underline{\boldsymbol{0}}\end{bmatrix}}_{\underline{\boldsymbol{B}}_{\mathrm{r}}} \underbrace{\begin{bmatrix}\boldsymbol{F}_{\mathrm{r}}\\ \boldsymbol{T}_{\mathrm{r}}\end{bmatrix}}_{\boldsymbol{u}_{\mathrm{r}}}. \quad (1)$$

The state vector consists of the total CoM translational ($\boldsymbol{v}_{\mathrm{r}}$) and rotational ($\boldsymbol{\omega}_{\mathrm{r}}$) velocities, expressed in the rover frame, as well as the total Euler angles of the rover $\boldsymbol{\theta}$, following the Roll-Pitch-Yaw (RPY) sequence. The input to the system is the force torque vector $[\boldsymbol{F}_{\mathrm{r}}, \boldsymbol{T}_{\mathrm{r}}]^\top$ of size $\mathbb{R}^{(6\times1)}$ with the constant input matrix consisting of the inverted mass and inertia matrices. $\underline{\boldsymbol{S}}$ is the cross-product building matrix (see [33, p. 107]), ${}^{\mathrm{r}}\underline{\boldsymbol{R}}_0$ is the rotation matrix that transforms a vector from the 0-frame to the r-frame and ${}^0\boldsymbol{g} = [0, 0, -9.81]^\top$ is the gravity vector expressed in the 0-frame. Given the definition of the RPY angles $\boldsymbol{\theta} = [\phi, \vartheta, \psi]^\top$, their dynamics need the matrix

$$\underline{\boldsymbol{U}}^{-1} = \frac{1}{\cos(\vartheta)}\begin{bmatrix}\cos(\vartheta) & \sin(\phi)\sin(\vartheta) & \cos(\phi)\sin(\vartheta)\\ 0 & \cos(\phi)\cos(\vartheta) & -\sin(\phi)\cos(\vartheta)\\ 0 & \sin(\phi) & \cos(\phi)\end{bmatrix}. \quad (2)$$

The nonlinear acceleration in equation (1) consists of the gravitational acceleration and the cross-product of rotational and translational velocity due to the describing rover-frame (r-frame) not being fixed in space.

2.2 Rover (differential) kinematics

While a lumped mass in one rigid body was assumed for the dynamics in section 2.1, this section deals with the actual passive suspension system of the ExoMars rover, its kinematics and differential kinematics. The approach to derive these is well known in robotics and can, e. g., be found in the book of Siciliano et al. [33].

The rover has three passive bogies and six active steering joints, all of which are rotational degrees of freedom (DoF), leading to a vector $\boldsymbol{q} \in \mathbb{R}^{(9\times1)}$ of all DoF angles. In [5], the homogeneous transformation matrices, including the position vectors ${}^{\mathrm{r}}\boldsymbol{r}_{\mathrm{c}_i|\mathrm{r}}$ from the rover CoM to the contact points, were derived with the help of the *Denavit-Hartenberg (DH)* convention. The relative velocity of the ith contact frame can be computed by differentiation as

$$\boldsymbol{v}_{\mathrm{c}_i|\mathrm{r}} = \frac{\partial \boldsymbol{r}_{\mathrm{c}_i|\mathrm{r}}}{\partial \boldsymbol{q}}\dot{\boldsymbol{q}} = \underline{\boldsymbol{J}}_{i,\mathrm{rel}}(\boldsymbol{q})\dot{\boldsymbol{q}}, \tag{3}$$

defining the linear velocity part of the relative geometric Jacobian $\underline{\boldsymbol{J}}_{i,\mathrm{rel}}$. The linear velocity of the r-frame and the cross-product of its angular velocity with the distance to the contact frame must be added to the relative velocity (3) due to the moving r-frame. Additionally, the total velocity needs to be expressed in the contact frame which is achieved with an additional rotation ${}^{\mathrm{c}_i}\underline{\boldsymbol{R}}_{\mathrm{r}}$, leading to the ith wheel total velocity

$${}^{\mathrm{c}_i}\boldsymbol{v}_{\mathrm{c}_i|0} = \underbrace{{}^{\mathrm{c}_i}\underline{\boldsymbol{R}}_{\mathrm{r}}(\boldsymbol{q})\begin{bmatrix}\underline{\boldsymbol{I}} & -\underline{\boldsymbol{S}}(\boldsymbol{r}_{\mathrm{c}_i|\mathrm{r}}) & \underline{\boldsymbol{J}}_{i,\mathrm{rel}}(\boldsymbol{q})\end{bmatrix}}_{=\underline{\boldsymbol{J}}_i(\boldsymbol{q})}\begin{bmatrix}{}^{\mathrm{r}}\boldsymbol{v}_{\mathrm{r}|0}\\ {}^{\mathrm{r}}\boldsymbol{\omega}_{\mathrm{r}|0}\\ \dot{\boldsymbol{q}}\end{bmatrix}. \tag{4}$$

The force and torque equilibrium equation that relates contact forces to rover forces can now be established through the *kineto-static duality* [33, p. 148] as

$$\begin{bmatrix}\boldsymbol{F}_{\mathrm{r}}\\ \boldsymbol{T}_{\mathrm{r}}\\ \boldsymbol{\tau}_q\end{bmatrix} = \underline{\boldsymbol{J}}^{\top}(\boldsymbol{q})\boldsymbol{F}_{\mathrm{c}} \quad \text{with} \quad \underline{\boldsymbol{J}}(\boldsymbol{q}) = \begin{bmatrix}\underline{\boldsymbol{J}}_1(\boldsymbol{q})\\ \vdots\\ \underline{\boldsymbol{J}}_6(\boldsymbol{q})\end{bmatrix}. \tag{5}$$

Therefore it is assumed that only forces and no torques can be transferred in the contact points.

2.3 Combined wheel and contact dynamics

On individual wheel level, $n = 6$ identical dynamic models are developed. These contain the wheel dynamics with its inertia as well as a quasi-static estimation of normal forces and a simple wheel-ground contact model for the tangential forces.

Quasi-static estimation of normal forces

Normal force models for soft soil contact exist but rely on a sinkage of the wheel, as well as many parameters that are hard to identify. Due to the difficulties of modeling these forces accurately [4], the lack of a sinkage measurement and the slow movement of the robot – the maximum velocity is about $0.03\,\frac{\mathrm{m}}{\mathrm{s}}$ – the normal forces are estimated with a quasi-static force and torque balance.

Three force/torque balance equations are set up for the calculation. A force equilibrium between the gravitational force acting in the CoM and the force component in the direction of gravity at each bogie reads

$$\sum_{j=1}^{3} F_{\mathrm{g,b}_j} - gm = 0 \tag{6}$$

with the gravitational force gm and the force in the direction of the gravity at the jth bogie $F_{\mathrm{g,b}_j}$. Another force equilibrium is stated for each of the three bogies as

$$F_{\mathrm{g,b}_j} - F_{\mathrm{g,c}_{j_1}} - F_{\mathrm{g,c}_{j_2}} = 0, \quad j = 1\ldots3 \tag{7}$$

where the wheels j_1 and j_2 are attached to the jth bogie. Torque sums around the x- and y-axes of the 0-frame must be zero due to the quasi-static assumption,

$$\begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\end{bmatrix}{}^{0}\underline{\boldsymbol{R}}_{\mathrm{r}}\sum_{j=1}^{3}\left(\boldsymbol{r}_{\mathrm{b}_j|\mathrm{r}} \times {}^{\mathrm{r}}\boldsymbol{e}_{z_0}\right)F_{\mathrm{g,b}_j} = \boldsymbol{0}, \tag{8}$$

with the unit vector of the global z-axis (the gravity direction) expressed in the r-frame ${}^{\mathrm{r}}\boldsymbol{e}_{z_0}$. The torque around each bogie's joint axis must be zero for zero angular acceleration because the bogies are ideal free rotating joints and thus can not transfer torques,

$$\begin{aligned}[0 \quad 0 \quad 1]\,{}^{\mathrm{b}_j}\underline{\boldsymbol{R}}_{\mathrm{r}}\big({}^{\mathrm{r}}\boldsymbol{r}_{\mathrm{c}_{j_1}|\mathrm{b}_j} \times {}^{\mathrm{r}}\boldsymbol{e}_{z_0}F_{\mathrm{g,c}_{j_1}}\\ + {}^{\mathrm{r}}\boldsymbol{r}_{\mathrm{c}_{j_2}|\mathrm{b}_j} \times {}^{\mathrm{r}}\boldsymbol{e}_{z_0}F_{\mathrm{g,c}_{j_2}}\big) = 0.\end{aligned} \tag{9}$$

Equations (6) to (9) add up to nine linearly independent scalar equations, which can be solved for the $6+3 = 9$ unknown scalar forces at the bogies and contact points. The scalars $F_{\mathrm{g,c}_i}$, $i = 1\ldots6$ all act in the direction of the

gravity, leading to the vector of the gravitational force at the ith contact point,

$$\boldsymbol{F}_{\mathrm{g},i} = F_{\mathrm{g,c}_i}\boldsymbol{e}_{z_0}. \tag{10}$$

Ultimately, only the component of $\boldsymbol{F}_{\mathrm{g},i}$ in the ground normal direction – i. e., the z-axis of the c-frame – is the searched contact normal force,

$$F_{\mathrm{N}_i}(\boldsymbol{\theta}_\mathrm{r}, \boldsymbol{q}_\mathrm{b}) = \langle \boldsymbol{e}_{z_0}, \boldsymbol{e}_{z_{\mathrm{c}_i}} \rangle F_{\mathrm{g,c}_i}. \tag{11}$$

Therein, $\langle \boldsymbol{e}_{z_0}, \boldsymbol{e}_{z_{\mathrm{c}_i}} \rangle$ is the scalar product of the two unit vectors and represents the projection of the ith gravitational force onto the contact frame's z-axis, see figure 2.

Tangential force

The wheel-ground interaction force is a combination of wheel-sand friction and sand-internal behavior. Planetary exploration rover wheels usually have *grousers* to achieve form lock instead of pure friction between wheel and sand in longitudinal wheel direction. Therefore, the maximum longitudinal force is assumed analogously to the Mohr-Coulomb maximum shear stress,

$$F_{\text{Mohr-Coulomb}} = F_{\mathrm{N}_i} \tan(\phi_{\mathrm{S-S}}), \tag{12}$$

where the internal sand shear angle $\phi_{\mathrm{S-S}}$ is in a range between 25° to 45° [38, 1] for sandy soil and cohesion can be neglected for the assumed dry sand [38].

This traction force acts opposed to the slip velocity of the contact point, which is the difference between the wheel rim velocity and the translational velocity of the contact point. The latter is taken from the differential kinematics (4) by dividing the joint angles $\boldsymbol{q}$ into the bogie angles $\boldsymbol{q}_\mathrm{b}$ and the steering angles $\boldsymbol{\delta}$:

$${}^{\mathrm{c}_i}\boldsymbol{v}_{\mathrm{c}_i|0} = {}^{\mathrm{c}_i}\underline{\boldsymbol{R}}_{\mathrm{s}_i}(\delta_i)\,{}^{\mathrm{s}_i}\boldsymbol{v}_{\mathrm{c}_i|0}(\boldsymbol{x}_\mathrm{r}, \boldsymbol{q}_\mathrm{b}, \dot{\boldsymbol{q}}_\mathrm{b}). \tag{13}$$

With the unit vector $\boldsymbol{e}_x = [1,0,0]^\top$, this leads to the slip velocity of the ith wheel

$$v_{\mathrm{sl},i} = \omega_i r - \boldsymbol{e}_x^{\top\,\mathrm{c}_i}\underline{\boldsymbol{R}}_{\mathrm{s}_i}(\delta_i)\,{}^{\mathrm{s}_i}\boldsymbol{v}_{\mathrm{c}_i|0}. \tag{14}$$

For friction phenomena, it is common to model the force direction opposing to the slip velocity direction with a signum function. The contact forces for a rigid wheel with loose sand, however, is much more complex and simplified models (see [38, 9]) depend on slip length and/or slip velocity. For this work, the actual traction force is modeled by a hyperbolic tangent with an appropriate parameter k_{vx},

$$F_{\mathrm{c}_i,x} = F_{\mathrm{N}_i} \tan(\phi_{\mathrm{S-S}}) \tanh(k_{vx} v_{\mathrm{sl},i}). \tag{15}$$

For the lateral wheel direction, pure metal-sand friction with a friction angle of $\phi_{\mathrm{M-S}} = 15°$ and a slip velocity parameter k_{vy} are assumed, yielding with $\boldsymbol{e}_y = [0,1,0]^\top$

$$F_{\mathrm{c}_i,y} = -F_{\mathrm{N}_i} \tan(\phi_{\mathrm{M-S}}) \tanh(k_{vy} \boldsymbol{e}_y^{\top\,\mathrm{c}_i}\underline{\boldsymbol{R}}_{\mathrm{s}_i}(\delta_i)\,{}^{\mathrm{s}_i}\boldsymbol{v}_{\mathrm{c}_i|0}). \tag{16}$$

More complex dynamic models can be integrated, although they might necessitate an adaption of the control method that will be chosen in section 3. More research on empirical terramechanics models, that are able to represent the dynamic behavior of the traction force and not just the steady state potential, is currently being done at the DLR Institute of System Dynamics and Control, cf. [4].

Wheel dynamics

The wheel assembly, in this work, consists of a drive and a steering actuator, which are both assumed ideal, i. e., the torque is realized exactly. Drive and steering rotational acceleration are thus computed with the respective input torques $T_{\omega,i}, T_{\delta,i}$ and the resistance force/torque from the ground contact. The steering angle is introduced as an additional state because it is needed for computation of the wheel forces and their transformation into the non-steered frame, see equations (14) to (16). These considerations result in the wheel dynamics equation

$$\begin{aligned}
\underbrace{\begin{bmatrix}\dot{\omega}_i\\ \dot{\delta}_i\\ \ddot{\delta}_i\end{bmatrix}}_{\boldsymbol{x}_{\mathrm{w}_i}} &= \underbrace{\begin{bmatrix}-\frac{r}{I_{\mathrm{w}_i,y}}F_{\mathrm{c}_i,x}\\ \dot{\delta}_i\\ -\frac{1}{I_{\mathrm{w}_i,z}}T_{\mathrm{res},z}\end{bmatrix}}_{\boldsymbol{f}_{\mathrm{w}_i}} + \underbrace{\begin{bmatrix}\frac{1}{I_{\mathrm{w}_i,y}} & 0\\ 0 & 0\\ 0 & \frac{1}{I_{\mathrm{w}_i,z}}\end{bmatrix}}_{\underline{\boldsymbol{B}}_{\mathrm{w}_i}} \underbrace{\begin{bmatrix}T_{\omega,i}\\ T_{\delta,i}\end{bmatrix}}_{\boldsymbol{u}_{\mathrm{w}_i}}\\
\underbrace{\begin{bmatrix}{}^{\mathrm{s}}F_{\mathrm{c}_i,x}\\ {}^{\mathrm{s}}F_{\mathrm{c}_i,y}\end{bmatrix}}_{\boldsymbol{y}_{\mathrm{w}_i}} &= \underbrace{{}^{\mathrm{s}}\underline{\boldsymbol{R}}_{\mathrm{w}}\begin{bmatrix}F_{\mathrm{c}_i,x}\\ F_{\mathrm{c}_i,y}\end{bmatrix}}_{\boldsymbol{h}_{\mathrm{w}_i}}
\end{aligned} \tag{17}$$

with the longitudinal $F_{\mathrm{c}_i,x}$ and lateral contact force $F_{\mathrm{c}_i,y}$ from equations (15) and (16), the steering resistance torque $T_{\mathrm{res},z}$ and the wheel inertia about the drive and steering axis $I_{\mathrm{w}_i,y}$ and $I_{\mathrm{w}_i,z}$, respectively. Note that the output $\boldsymbol{y}_{\mathrm{w}_i}$ contains the x- and y-force resolved in the non-steered s-frame in order to comply with the contact forces that are used in the force torque balance in equation (5) with

$$\boldsymbol{F}_\mathrm{c} = \begin{bmatrix}\boldsymbol{F}_{\mathrm{c}_1}\\ \vdots\\ \boldsymbol{F}_{\mathrm{c}_6}\end{bmatrix} \quad \text{and} \quad \boldsymbol{F}_{\mathrm{c}_i} = \begin{bmatrix}{}^{\mathrm{s}}F_{\mathrm{c}_i,x}\\ {}^{\mathrm{s}}F_{\mathrm{c}_i,y}\\ F_{\mathrm{N}_i}\end{bmatrix}. \tag{18}$$

3 Controller synthesis

Corresponding to the modeling approach, the control system consists of three sub-controllers: First, the force trajec-

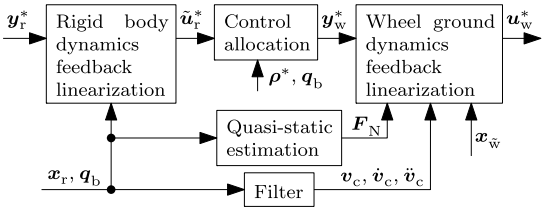

Figure 3: Layout of the proposed controller.

tory on full rover level is computed from the velocity trajectory and rover states via feedback linearization of the rigid body dynamics model from section 2.1. Second, the rover-level forces are distributed to wheel forces via control allocation based on the rover kinematics from section 2.2 and a nonlinear optimization. Lastly, an underlying single-wheel controller is synthesized via feedback linearization of the wheel and contact dynamics model from section 2.3. The overall architecture, with its in- and outputs, can be seen in figure 3.

3.1 The method of feedback linearization

The method of feedback linearization [19] is used for the rigid body and ground contact dynamics from sections 2.1 and 2.3, because the method is well suited for trajectory tracking problems of nonlinear systems [21]. This technique is briefly summarized here for a tidy controller synthesis in the following sections; for details refer to [19].

For an input-affine MIMO system

$$\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{f}(\boldsymbol{x}) + \underline{\boldsymbol{g}}(\boldsymbol{x})\boldsymbol{u} \\ \boldsymbol{y} &= \boldsymbol{h}(\boldsymbol{x}), \end{aligned} \tag{19}$$

the p_ith derivative of each of the n_y outputs is computed such that it is a function of at least one of the n_u inputs and the $(p_i - 1)$th derivative is not a function of any of the inputs. The *vector relative degree* then results as

$$\boldsymbol{p} = \begin{bmatrix} p_1 & \dots & p_{n_y} \end{bmatrix}^\top \tag{20}$$

if the matrix

$$\underline{\boldsymbol{V}} = \begin{bmatrix} L_{\boldsymbol{g}_1} L_{\boldsymbol{f}}^{p_1-1} h_1(\boldsymbol{x}) & \dots & L_{\boldsymbol{g}_{n_u}} L_{\boldsymbol{f}}^{p_1-1} h_1(\boldsymbol{x}) \\ \vdots & \ddots & \vdots \\ L_{\boldsymbol{g}_1} L_{\boldsymbol{f}}^{p_{n_y}-1} h_{n_y}(\boldsymbol{x}) & \dots & L_{\boldsymbol{g}_{n_u}} L_{\boldsymbol{f}}^{p_{n_y}-1} h_{n_y}(\boldsymbol{x}) \end{bmatrix} \tag{21}$$

has full rank with the *Lie-derivative*

$$L_{\boldsymbol{f}} \boldsymbol{h}(\boldsymbol{x}) = \frac{\partial \boldsymbol{h}(\boldsymbol{x})}{\partial \boldsymbol{x}} \boldsymbol{f}(\boldsymbol{x}) \tag{22}$$

and the kth Lie-derivative

$$L_{\boldsymbol{f}}^k \boldsymbol{h}(\boldsymbol{x}) = \frac{\partial L_{\boldsymbol{f}}^{k-1} \boldsymbol{h}(\boldsymbol{x})}{\partial \boldsymbol{x}} \boldsymbol{f}(\boldsymbol{x}). \tag{23}$$

Asymptotic feedback tracking of a trajectory $\boldsymbol{y}^*, \dot{\boldsymbol{y}}^*, \dots, \boldsymbol{y}^{*(\boldsymbol{p})}$ is then achieved with the control law

$$\begin{aligned} \boldsymbol{u}^* = \underline{\boldsymbol{V}}^{-1} \Bigg(& - \begin{bmatrix} L_{\boldsymbol{f}}^{p_1} h_1 \\ \vdots \\ L_{\boldsymbol{f}}^{p_{n_y}} h_{n_y}, \end{bmatrix} + \begin{bmatrix} y_1^{*(p_1)} \\ \vdots \\ y_{n_y}^{*(p_{n_y})} \end{bmatrix} \\ & - \begin{bmatrix} \sum_{i_1=1}^{p_1} k_{1,i_1-1} \left(L_{\boldsymbol{f}}^{i_1-1} h_1 - y_1^{*(i_1-1)} \right) \\ \vdots \\ \sum_{i_{n_y}=1}^{p_{n_y}} k_{n_y,i_{n_y}-1} \left(L_{\boldsymbol{f}}^{i_{n_y}-1} h_{n_y} - y_1^{*(i_{n_y}-1)} \right) \end{bmatrix} \Bigg). \end{aligned} \tag{24}$$

Therein $\boldsymbol{k}_j^\top = [k_{j,0}, \dots, k_{j,p_j-1}]$ with $j = 1 \dots n_y$ are the controller gains for the linearized and decoupled closed loop dynamics.

The closed-loop system with equations (19) and (24) results in chains of p_i integrators for the ith output and therefore a system with $\tilde{n} = \sum_{i=1}^{n_y} p_i$ coordinates $\boldsymbol{\xi}$. These new coordinates consist of the outputs and their 1st to $(p_i - 1)$th derivatives. If the number of new coordinates is lower than the order of the open loop system (19) $\tilde{n} < n$, there are $n - \tilde{n}$ hidden states $\boldsymbol{\eta}$. Finding these states is, in general, not straightforward, see [19], and their stability must be investigated with the *zero dynamics* by setting all outputs and output derivatives (i. e., the closed-loop coordinates $\boldsymbol{\xi}$) to zero.

3.2 Rigid body dynamics feedback linearization

Model modifications

The chassis control for mobile robots usually receives forward, sideways and yaw velocity trajectories in the rover frame (see figure 1),

$$\boldsymbol{y}_r^* = \begin{bmatrix} v_x^* \\ v_y^* \\ \omega_z^* \end{bmatrix}. \tag{25}$$

To proceed, the rover force and torque equilibrium from equation (5) is considered and must be further di-

vided into two parts:

$$\begin{bmatrix} \boldsymbol{F}_\mathrm{r} \\ \boldsymbol{T}_\mathrm{r} \end{bmatrix} = \begin{bmatrix} \boldsymbol{F}_\mathrm{r,N} \\ \boldsymbol{T}_\mathrm{r,N} \end{bmatrix} + \begin{bmatrix} \boldsymbol{F}_\mathrm{r,T} \\ \boldsymbol{T}_\mathrm{r,T} \end{bmatrix} = \underline{\boldsymbol{J}}_{F,\mathrm{N}}^\top(\boldsymbol{q}_\mathrm{b})\boldsymbol{F}_\mathrm{c,N}(\boldsymbol{q}_\mathrm{b}, \boldsymbol{x}_\theta) + \underline{\boldsymbol{J}}_{F,\mathrm{T}}^\top(\boldsymbol{q}_\mathrm{b})\boldsymbol{F}_\mathrm{c,T}. \tag{26}$$

The first part (index 'N') are the normal forces at the contact patch and the associated columns of the transposed force Jacobian, while the second part (index 'T') considers only the tangential contact forces.

Since the ExoMars suspension is passive, the normal contact forces cannot be actively controlled. Therefore the first summand in equation (26) is added to the nonlinear dynamics of equation (1), leading to the modified nonlinear expression

$$\tilde{\boldsymbol{f}}_\mathrm{r}(\boldsymbol{x}_\mathrm{r}, \boldsymbol{q}_\mathrm{b}) = \boldsymbol{f}_\mathrm{r}(\boldsymbol{x}_\mathrm{r}) + \underline{\boldsymbol{B}}_\mathrm{r}\underline{\boldsymbol{J}}_{F,\mathrm{N}}^\top(\boldsymbol{q}_\mathrm{b})\boldsymbol{F}_\mathrm{N}(\boldsymbol{q}_\mathrm{b}, \boldsymbol{x}_\mathrm{r}), \tag{27}$$

where the bogie angles are treated as a measurable disturbance and the normal forces are estimated through the quasi-stationary force-torque balance (11).

Due to the kinematics of the rover, the rover forces resulting from tangential wheel forces $[\boldsymbol{F}_\mathrm{r,T}, \boldsymbol{T}_\mathrm{r,T}]^\top$ all act in the rover's x–y-plane in the nominal configuration (i. e., bogie angles $\boldsymbol{q}_\mathrm{b} = 0$). Hence, the force in its z direction and the torques around its x and y axes are not controllable in the nominal configuration, which is reflected in a rank deficiency of the transposed Jacobian $\underline{\boldsymbol{J}}_{F,\mathrm{T}}^\top(\boldsymbol{q}_\mathrm{b} = \boldsymbol{0})$ in equation (26). To guarantee a full rank of the allocation matrix for all configurations with bogie angles smaller than 90°, which is needed for the controller synthesis, the input vector

$$\tilde{\boldsymbol{u}}_\mathrm{r} = \begin{bmatrix} F_{\mathrm{r},x} \\ F_{\mathrm{r},y} \\ T_{\mathrm{r},z} \end{bmatrix} \tag{28}$$

is established. Bogie angles of 90° or higher are to be prevented anyways, since the rover would loose its stability. In comparison to the original input vector $\boldsymbol{u}_\mathrm{r}$ from equation (1), the forces in vertical direction and the roll and pitch torque are not considered in $\tilde{\boldsymbol{u}}_\mathrm{r}$ and are treated as disturbance for the synthesis

$$\boldsymbol{z}_F = \begin{bmatrix} F_{\mathrm{r},z} \\ T_{\mathrm{r},x} \\ T_{\mathrm{r},y} \end{bmatrix}. \tag{29}$$

Even though these disturbance force and torques are known through equation (26), they are neglected in the controller synthesis. Since they have a higher differential order than the inputs, they do not show in the feedback linearization. A disturbance rejection could be considered in future work, but is expected to have little effect.

Ultimately, the rigid body dynamic model for synthesis of the feedback linearization controller results as

$$\begin{aligned} \dot{\boldsymbol{x}}_\mathrm{r} &= \tilde{\boldsymbol{f}}_\mathrm{r}(\boldsymbol{x}_\mathrm{r}, \boldsymbol{z}_\mathrm{b}) + \tilde{\underline{\boldsymbol{B}}}_\mathrm{r}\tilde{\boldsymbol{u}}_\mathrm{r} + \underline{\boldsymbol{E}}_\mathrm{r}\boldsymbol{z}_F, \\ \boldsymbol{y}_\mathrm{r} &= \underline{\boldsymbol{C}}_\mathrm{r}\boldsymbol{x}_\mathrm{r}, \end{aligned} \tag{30}$$

with $\tilde{\underline{\boldsymbol{B}}}_\mathrm{r}$ and $\underline{\boldsymbol{E}}_\mathrm{r}$ the respective columns of $\underline{\boldsymbol{B}}_\mathrm{r}$, $\tilde{\boldsymbol{f}}_\mathrm{r}$, $\tilde{\boldsymbol{u}}_\mathrm{r}$ and $\boldsymbol{z}_F$ from equations (27) to (29) and

$$\underline{\boldsymbol{C}}_\mathrm{r} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}. \tag{31}$$

Control law

For the rigid body dynamics system (30), the vector relative degree from equation (20) yields

$$\boldsymbol{p}_\mathrm{r} = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}^\top \tag{32}$$

and the linear input dependency and linear output equation lead to a constant matrix

$$\underline{\boldsymbol{V}}_\mathrm{r} = \underline{\boldsymbol{C}}_\mathrm{r}\tilde{\underline{\boldsymbol{B}}}_\mathrm{r} \tag{33}$$

with full rank. The coordinates are

$$\boldsymbol{\xi}_\mathrm{r} = \begin{bmatrix} v_x & v_y & \omega_z \end{bmatrix}^\top \tag{34}$$

and the control law for asymptotic tracking of the desired $\boldsymbol{y}_\mathrm{r}^*$ follows as

$$\begin{aligned} \tilde{\boldsymbol{u}}_\mathrm{r}^* = \underline{\boldsymbol{V}}_\mathrm{r}^{-1}\Big(&- \underline{\boldsymbol{C}}_\mathrm{r}\tilde{\boldsymbol{f}}_\mathrm{r} + \dot{\boldsymbol{y}}_\mathrm{r}^* - \underline{\boldsymbol{K}}_{\mathrm{r},0}\left(\boldsymbol{y}_\mathrm{r} - \boldsymbol{y}_\mathrm{r}^*\right) \\ &- \underline{\boldsymbol{K}}_{\mathrm{r},\mathrm{I}}\int \left(\boldsymbol{y}_\mathrm{r} - \boldsymbol{y}_\mathrm{r}^*\right)\mathrm{d}t\Big). \end{aligned} \tag{35}$$

An additional integrator is used for compensation of unknown disturbances with the integrator gain matrix $\underline{\boldsymbol{K}}_{\mathrm{r},\mathrm{I}}$.

Stability of the zero dynamics

The order of the closed-loop system (30) with (35) is $\sum_{i=1}^{n_y} p_i = 3$ compared to the open loop system order of equation (30), $n = 9$. As described in section 3.1, this leads to six hidden states

$$\boldsymbol{\eta}_\mathrm{r} = \begin{bmatrix} v_z & \omega_x & \omega_y & \phi & \vartheta & \psi \end{bmatrix}^\top, \tag{36}$$

that are, in this case, easy to find since the outputs and therewith the transformed coordinates $\boldsymbol{\xi}_\mathrm{r}$ are a subset of the states. For the analysis of the zero dynamics, the normal force must be modeled dependent on the rover position and orientation instead of being estimated through

the quasi-stationary force-torque balance, as it was used in equation (27). A simple and accurate enough model is a penetration-based spring-damper model, which uses the sinkage s_i of the ith wheel (see figure 2)

$$s_i = z_{g,i} - \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \left(\boldsymbol{r}_{r|0} + {}^{0}\underline{\boldsymbol{R}}_{r}(\boldsymbol{\theta}) \boldsymbol{r}_{c_i|r}(\boldsymbol{q}_b) \right) \tag{37}$$

and its derivative $\dot{s}_i$. The unknown disturbance $z_{g,i}$ is the soil height at the ith wheel expressed in the global 0-frame and $\dot{z}_{g,i}$, its derivative. The rover position $\boldsymbol{r}_{r|0}$ is expressed in the 0-frame as well and $\boldsymbol{r}_{c_i|r}(\boldsymbol{q}_b)$ is the vector from the rover frame (r) to the ith contact frame (c).

It can be seen in equation (37), that only the vertical z-component of the rover position $r_{r|0,z}$ is needed for the sinkage calculation. With the spring and damper constant, c and d, respectively, the ith wheel normal force results in

$$F_{N_i}(\boldsymbol{x}_r, r_{r|0,z}, \boldsymbol{q}_b, \boldsymbol{z}_g, \dot{\boldsymbol{z}}_g) = cs_i + d\dot{s}_i. \tag{38}$$

The derivative of the rover's vertical position is calculated through the rover velocity with

$$\dot{r}_{r|0,z} = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} {}^{0}\underline{\boldsymbol{R}}_{r} \boldsymbol{x}_v, \tag{39}$$

hence the z-position is added to $\boldsymbol{\eta}_r$ as additional state. For the zero dynamics analysis, the outputs, and thereby the transformed states, are set to zero $\boldsymbol{\xi}_r = \boldsymbol{0}$, which results in

$$\dot{\tilde{\boldsymbol{\eta}}}_r = \begin{bmatrix} \dot{\boldsymbol{\eta}}_r \\ \dot{\boldsymbol{r}}_{r|0} \end{bmatrix} = \boldsymbol{g}(\tilde{\boldsymbol{\eta}}_r, \boldsymbol{q}_b, \boldsymbol{z}_g, \dot{\boldsymbol{z}}_g) \tag{40}$$

Since a Lyapunov function for the nonlinear system (40) could not be found, it is linearized around its equilibrium in the origin $\tilde{\boldsymbol{\eta}}_r = \boldsymbol{0}$, yielding

$$\dot{\tilde{\boldsymbol{\eta}}}_r = \underline{\boldsymbol{A}}_{lin}(\tilde{\boldsymbol{\eta}}_r, \boldsymbol{z}_b) \tilde{\boldsymbol{\eta}}_r. \tag{41}$$

Note that an origin in $\tilde{\boldsymbol{\eta}}_r = \boldsymbol{0}$ can only be achieved with a specific value of the disturbances. For this work the bogie angles are chosen zero and the soil height $\boldsymbol{z}_g$ is found by solving the equilibrium condition $\dot{\tilde{\boldsymbol{\eta}}}_r(\tilde{\boldsymbol{\eta}}_r = \boldsymbol{0}) = \boldsymbol{0}$ with $\dot{\boldsymbol{z}}_g = \boldsymbol{0}$ for $\boldsymbol{z}_g$.

In an eigenvalue (EV) analysis, one of the seven EVs of the linearization matrix $\underline{\boldsymbol{A}}_{lin}$ shows a zero real part, which, according to Lyapunov's indirect method, does not yield a conclusion about the stability of the origin. However, further analysis of $\underline{\boldsymbol{A}}_{lin}$ has shown that the row and column associated with the yaw angle of the rover are zero. This means that the zero dynamics of the yaw angle is equal to zero, $\dot{\psi} = 0$, and the value of it does not have an influence on any of the other states' dynamics. By removing the yaw angle dynamics from the linearized equation (41), the zero row and column vanish from $\underline{\boldsymbol{A}}_{lin}$ and only EVs in the negative complex half plane result, which yields stability of the origin according to Lyapunov's indirect method. The found zero EV of the yaw angle can be explained figuratively by a lack of an angle-based yaw torque as opposed to the other angles and the vertical rover position, which are affected by the position-based spring normal forces. In other words, while the yaw velocity is one of the stabilized $\boldsymbol{\xi}_r$-states, the yaw angle can be arbitrary, which, however, does not influence the stability of the origin for any of the other states.

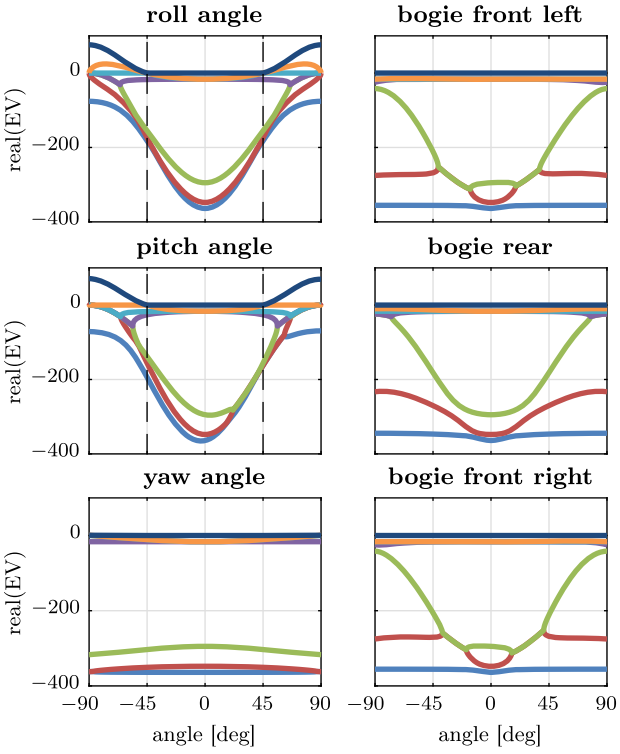

Figure 4: Real parts of the zero dynamics' eigenvalues for variation of one angle at a time. Each line represents one of the seven EVs. The axes labels and the scaling are identical for all subplots.

A variation of the zero dynamics states $\tilde{\boldsymbol{\eta}}_r$ and bogie angles was performed, where only one state or bogie angle was varied at a time while keeping all others at zero. Figure 4 shows the results, where each line corresponds to one EV. The vertical, pitch and roll velocities do not influence the real parts of the EVs and are therefore not shown. It can be seen that the absolute pitch and roll angles must stay below 45° while requirements on all other angles are less restrictive – ±90° poses no problem. For a simultaneous variation of multiple angles at a time, there are combinations within the implied ranges that lead to instabilities. A full variation has shown that the EVs – besides the one zero EV, see above – of the linearized zero dynamics are

in the left complex half-plane for arbitrary state and bogie angle combinations in the ranges

$$\begin{aligned} -34° < \phi, \vartheta, \psi < 34° \\ -45° < z_{\mathrm{b},i} < 45° \end{aligned} \tag{42}$$

The angle ranges are within the expected stability range due to the height of the center of gravity with respect to the chassis width and length.

Although these consideration are not a rigorous proof of stability of the nonlinear zero dynamics, the findings from the linear analysis correspond well to physical reasoning and thus stability of the zero dynamics is concluded.

3.3 Wheel force allocation

Purpose of the force allocation is the computation of trajectories for the individual contact point tangential forces

$${}^{\mathrm{s}}\boldsymbol{F}^{*}_{\mathrm{c,T}} = {}^{\mathrm{s}}\begin{bmatrix} \boldsymbol{F}^{*}_{\mathrm{c}_1,\mathrm{T}} \\ \vdots \\ \boldsymbol{F}^{*}_{\mathrm{c}_6,\mathrm{T}} \end{bmatrix} \tag{43}$$

from the rover force trajectory $\tilde{\boldsymbol{u}}^{*}_{\mathrm{r}}$ (see equation (28)), where each tangential force consists of x- and y-direction

$${}^{\mathrm{s}}\boldsymbol{F}_{\mathrm{c}_i,\mathrm{T}} = \begin{bmatrix} {}^{\mathrm{s}}F_{\mathrm{c}_i,x} \\ {}^{\mathrm{s}}F_{\mathrm{c}_i,y} \end{bmatrix} = \boldsymbol{y}_{\mathrm{w}_i} \tag{44}$$

and is the output of the ith wheel ground system from equation (17). These contact forces are expressed in their respective non-steered s_i-frame since the steering and drive actuators are to be considered only in the underlying wheel-ground-dynamics controller, see section 3.4. Note that only the tangential contact forces are considered here, since the non-controllable normal forces were added to the rigid body dynamics in equation (27).

The considerations in sections 2.2 and 3.2 led to the partitioned force-torque balance equation (26). Extracting the rows of interest, the allocation matrix $\underline{\boldsymbol{G}}$ can be defined,

$$\tilde{\boldsymbol{u}}_{\mathrm{r}} = \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \boldsymbol{J}^{\top}_{F,\mathrm{T}}(\boldsymbol{q}_{\mathrm{b}})}_{=\underline{\boldsymbol{G}}} {}^{\mathrm{s}}\boldsymbol{F}_{\mathrm{c,T}}(\boldsymbol{q}_{\mathrm{b}}, \boldsymbol{x}_{\theta}). \tag{45}$$

For six contact points, $\underline{\boldsymbol{G}}$ has three rows and 12 columns, which means that the system of equations (45) is underdetermined, i. e., there are infinitely many solutions for the contact force trajectory ${}^{\mathrm{s}}\boldsymbol{F}^{*}_{\mathrm{c,T}}$, assuming that $\underline{\boldsymbol{G}}$ has full row rank. A common solution (see [15]) to handle such underdetermined allocation problems is to compute the pseudo-inverse $\underline{\boldsymbol{G}}^{+}$ and a basis of the null-space $\underline{\boldsymbol{G}}^{\perp}$ of the allocation matrix, yielding

$${}^{\mathrm{s}}\boldsymbol{F}^{*}_{\mathrm{c,T}} = \underline{\boldsymbol{G}}^{+}\tilde{\boldsymbol{u}}^{*}_{\mathrm{r}} + \underline{\boldsymbol{G}}^{\perp}\boldsymbol{\rho}^{*}. \tag{46}$$

Since, by definition of a null-space basis,

$$\underline{\boldsymbol{G}}\underline{\boldsymbol{G}}^{\perp}\boldsymbol{\rho} = \boldsymbol{0} \quad \forall \quad \boldsymbol{\rho} \in \mathbb{R}, \tag{47}$$

the degrees of freedom $\boldsymbol{\rho}^{*}$ in equation (46) can be chosen arbitrarily without changing the resulting rover force-torque vector $\tilde{\boldsymbol{u}}_{\mathrm{r}}$ in equation (45). While choosing $\boldsymbol{\rho}^{*} = \boldsymbol{0}$ in equation (46) yields the least-squares solution for the contact forces, $\boldsymbol{\rho}^{*}$ can be varied for different solutions.

In this work, the degrees of freedom are adjusted by the nonlinear optimization

$$\min \sum_{i=1}^{6} \left(\frac{\| {}^{\mathrm{s}}\boldsymbol{F}^{*}_{\mathrm{c}_i,\mathrm{T}}(\boldsymbol{\rho}) \|}{F_{N,i}} \right)^2 \tag{48}$$

which minimizes the quadratic sum of the relation between tangential and normal contact force. This sum of fractions yields a minimization of the traction potential usage. For soft sand, contact force models are known to have high uncertainty, especially for changing soil conditions, but they always depend on the normal force. Therefore, this objective function aims at staying away as far as possible from the maximum assumed contact force. Reaching or exceeding this maximum force leads to soil failure and extended, potentially dangerous, sinkage of the associated wheel. The optimization is solved with the *fmincon* solver in MATLAB which uses the interior-point algorithm.

Due to the nonlinear optimization setup, other objectives, e. g., energy optimal driving, are possible without any changes to the algorithm. To ensure computational efficiency, the optimization is not performed within the main control loop but outside with a much lower update frequency. The previous optimal set of DoFs $\boldsymbol{\rho}^{*}$ is used until a new one has been computed, which was found to be not critical, since the DoFs change slowly for slowly changing terrain and slope properties.

3.4 Wheel dynamics feedback linearization

To control the underlying single wheel systems, consisting of one drive and one steering actuator each, feedback linearization is applied again. The first input of the system (17), the drive torque, appears in the first derivative of both outputs, while the second input, the steering torque, only

appears in the second output derivatives. This yields a zero column in the 2×2 $\underline{\boldsymbol{V}}_{\mathrm{w}_i}$-matrix according to equation (21), i. e., it is singular and thus a feedback linearization is not possible.

To solve that problem, the wheel-ground system is modified with a *dynamic extension*[19, p. 249ff], which is performed on the drive torque input in this work. The input of the dynamically extended system reads

$$\boldsymbol{u}_{\tilde{\mathrm{w}}_i} = \begin{bmatrix} \dot{T}_{\omega,i} \\ T_{\delta,i} \end{bmatrix} = \begin{bmatrix} \dot{u}_{\mathrm{w}_i,1} \\ u_{\mathrm{w}_i,2} \end{bmatrix}. \tag{49}$$

To distinguish the dynamically extended wheel-ground dynamics from the original one, a tilded w-index $(\cdot)_{\tilde{\mathrm{w}}}$ is used. By adding the original first input $T_{\omega,i}$ as a state of the system, the modified state equation is

$$\underbrace{\begin{bmatrix} \dot{\omega}_i \\ \dot{\delta}_i \\ \ddot{\delta}_i \\ \dot{T}_{\omega,i} \end{bmatrix}}_{\dot{\boldsymbol{x}}_{\tilde{\mathrm{w}}_i}} = \underbrace{\begin{bmatrix} -\frac{rF_{\mathrm{c}_i,x}}{I_{\mathrm{w}_i,y}} + \frac{T_{\omega,i}}{I_{\mathrm{w}_i,y}} \\ \dot{\delta}_i \\ -\frac{1}{I_{\mathrm{w}_i,z}} T_{\mathrm{res},z} \\ 0 \end{bmatrix}}_{\boldsymbol{f}_{\tilde{\mathrm{w}}_i}} + \underbrace{\begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & \frac{1}{I_{\mathrm{w}_i,z}} \\ 1 & 0 \end{bmatrix}}_{\underline{\boldsymbol{B}}_{\tilde{\mathrm{w}}_i}} \boldsymbol{u}_{\tilde{\mathrm{w}}_i}, \tag{50}$$

while the output equation remains unchanged from equation (17). Note that the dynamics equation $\boldsymbol{f}_{\tilde{\mathrm{w}}_i}$ not only depends on the wheel states $\boldsymbol{x}_{\tilde{\mathrm{w}}_i}$ but through the wheel forces and the normal force estimation on the absolute velocity of the steering system from equation (13). Hereafter, this velocity is written as

$$\boldsymbol{v}_{\mathrm{c}_i} = {}^{\mathrm{s}_i}\boldsymbol{v}_{\mathrm{c}_i|0} \tag{51}$$

for better readability. Both, the velocity $\boldsymbol{v}_{\mathrm{c}_i}$ and the normal force F_{N_i}, depend on the rigid body states $\boldsymbol{x}_{\mathrm{r}}$, the bogie angles $\boldsymbol{q}_{\mathrm{b}}$ and their derivatives $\dot{\boldsymbol{q}}_{\mathrm{b}}$ according to equations (4) and (11).

While the normal forces F_{N_i} can be assumed piecewise constant, the velocity of the contact points $\boldsymbol{v}_{\mathrm{c}_i}$ needs to be considered as a general time-dependent quantity. This yields the first derivative of the output

$$\begin{aligned} \dot{\boldsymbol{y}}_{\tilde{\mathrm{w}}_i} &= \frac{\partial \boldsymbol{h}_{\tilde{\mathrm{w}}_i}}{\partial \boldsymbol{x}_{\tilde{\mathrm{w}}_i}} \dot{\boldsymbol{x}}_{\tilde{\mathrm{w}}_i} + \frac{\partial \boldsymbol{h}_{\tilde{\mathrm{w}}_i}}{\partial \boldsymbol{v}_{\mathrm{c}_i}} \dot{\boldsymbol{v}}_{\mathrm{c}_i} \\ &= L_{\boldsymbol{f}_{\tilde{\mathrm{w}}_i}} \boldsymbol{h}_{\tilde{\mathrm{w}}_i} + \underbrace{L_{\underline{\boldsymbol{B}}_{\tilde{\mathrm{w}}_i}} \boldsymbol{h}_{\tilde{\mathrm{w}}_i}}_{=\underline{\boldsymbol{0}}} \boldsymbol{u}_{\tilde{\mathrm{w}}_i} + \frac{\partial \boldsymbol{h}_{\tilde{\mathrm{w}}_i}}{\partial \boldsymbol{v}_{\mathrm{c}_i}} \dot{\boldsymbol{v}}_{\mathrm{c}_i}, \end{aligned} \tag{52}$$

with the Lie-derivative from equation (22). Since there is no input dependency in either of the output derivatives, the second derivative is computed as

$$\ddot{\boldsymbol{y}}_{\tilde{\mathrm{w}}_i} = \frac{\partial \dot{\boldsymbol{y}}_{\tilde{\mathrm{w}}_i}}{\partial \boldsymbol{x}_{\tilde{\mathrm{w}}_i}} \left(\boldsymbol{f}_{\tilde{\mathrm{w}}_i} + \underline{\boldsymbol{B}}_{\tilde{\mathrm{w}}_i} \boldsymbol{u}_{\tilde{\mathrm{w}}_i} \right) + \frac{\partial \dot{\boldsymbol{y}}_{\tilde{\mathrm{w}}_i}}{\partial \boldsymbol{v}_{\mathrm{c}_i}} \dot{\boldsymbol{v}}_{\mathrm{c}_i} + \frac{\partial \dot{\boldsymbol{y}}_{\tilde{\mathrm{w}}_i}}{\partial \dot{\boldsymbol{v}}_{\mathrm{c}_i}} \ddot{\boldsymbol{v}}_{\mathrm{c}_i} \tag{53}$$

In both outputs' second derivative, both inputs appear and equation (53) can be resorted to

$$\begin{aligned} \ddot{\boldsymbol{y}}_{\tilde{\mathrm{w}}_i} =& \boldsymbol{w}_{\tilde{\mathrm{w}}_i}(\boldsymbol{x}_{\tilde{\mathrm{w}}_i}, \boldsymbol{v}_{\mathrm{c}_i}, \dot{\boldsymbol{v}}_{\mathrm{c}_i}, \ddot{\boldsymbol{v}}_{\mathrm{c}_i}, F_{N_i}) \\ &+ \underline{\boldsymbol{V}}_{\tilde{\mathrm{w}}_i}(\boldsymbol{x}_{\tilde{\mathrm{w}}_i}, F_{N_i}) \boldsymbol{u}_{\tilde{\mathrm{w}}_i}. \end{aligned} \tag{54}$$

The $\underline{\boldsymbol{V}}_{\tilde{\mathrm{w}}_i}$-matrix and $\boldsymbol{w}_{\tilde{\mathrm{w}}_i}$-vector are the elements that are needed for the feedback linearization according to section 3.1. The $\underline{\boldsymbol{V}}_{\tilde{\mathrm{w}}_i}$-matrix is regular, which means that the dynamically extended wheel-ground system has a vector relative degree of

$$\boldsymbol{r}_{\tilde{\mathrm{w}}_i} = [2 \quad 2]. \tag{55}$$

With a system order of $n = 4$, this yields no zero dynamics and the input trajectory for asymptotic tracking reads

$$\begin{aligned} \boldsymbol{u}^*_{\tilde{\mathrm{w}}_i} =& \underline{\boldsymbol{V}}^{-1}_{\tilde{\mathrm{w}}_i} \left(\ddot{\boldsymbol{y}}^*_{\tilde{\mathrm{w}}_i} - \boldsymbol{w}_{\tilde{\mathrm{w}}_i} \right. \\ &\left. - \underline{\boldsymbol{K}}_{\mathrm{w}_i,1} \left(\dot{\boldsymbol{y}}_{\tilde{\mathrm{w}}_i} - \dot{\boldsymbol{y}}^*_{\tilde{\mathrm{w}}_i} \right) - \underline{\boldsymbol{K}}_{\mathrm{w}_i,0} \left(\boldsymbol{y}_{\tilde{\mathrm{w}}_i} - \boldsymbol{y}^*_{\tilde{\mathrm{w}}_i} \right) \right). \end{aligned} \tag{56}$$

The gain matrices $\underline{\boldsymbol{K}}_{\mathrm{w}_i,0}, \underline{\boldsymbol{K}}_{\mathrm{w}_i,1}$ ensure asymptotic tracking with the feedback of the output error and its first derivative, respectively. Ultimately, equation (49) is used to compute the trajectory of the original inputs by integration as

$$\boldsymbol{u}^*_{\mathrm{w}_i} = \begin{bmatrix} T_{\omega,i} \\ T_{\delta,i} \end{bmatrix} = \begin{bmatrix} \int u^*_{\tilde{\mathrm{w}}_i,1} \mathrm{d}t \\ u^*_{\tilde{\mathrm{w}}_i,2} \end{bmatrix}. \tag{57}$$

4 Results

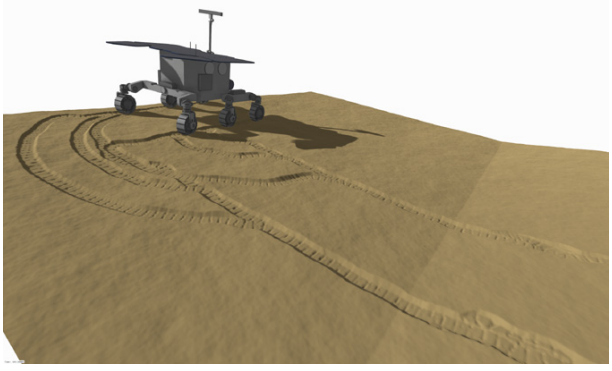

Figure 5: Screenshot of the ExoMars Rover, driving the presented scenario onto an inclined plane and performing a crab- and turn motion. The DLR Visualization Library [12] is attached to the multi-body simulation and the DLR Soil Contact Model [7] is used as soft soil contact model to create this 3D visualization. This is the disturbed case with non-nominal contact model and uneven ground, the plots of this drive are shown in figure 8.

For an evaluation of the presented approach, a co-simulation between the controller and a high-fidelity rover model is introduced. The rover model is built with the DLR Rover Simulation Toolkit [13] that is implemented in the multi-physics modeling language Modelica. Different fidelity models of the individual rover components can be selected which can include mechanical, electrical or even thermal model aspects. A sampled signal bus ensures realistic data exchange including sample and hold blocks that act in the set controller sampling interval. The proposed controller is implemented in MATLAB/Simulink with the use of the Symbolic Toolbox. Communication between Simulink and Modelica is set up with a blocking UDP connection which ensures that each model waits for execution of the other one.

In the test scenario, see figure 5, the ExoMars starts on a flat plane and is commanded to drive forward onto a 5° inclination. Once it is fully on the inclined plane, it is commanded forward, sideways and yaw velocities subsequently until it has turned 90° and continues purely forward. Note that the visualization in figure 5 shows the robustness test, see below, i. e., ground roughness and a non-nominal contact model with soil deformation are shown in contrast to this first, nominal, case. The overall geometry with its 0° and 5° parts, as well as the driven velocity trajectories are, however, the same in the nominal and the robustness test. Looking at the velocity trajectory (dashed bright lines in figure 6), the scenario is divided into 5 phases which are indicated in the figure:

- Rover starts to drive purely forward on 0° plane with $v_x^* = 0.03\,\frac{\mathrm{m}}{\mathrm{s}}$.
- Rover hits the 5° slope with front, middle, rear wheel pair consecutively.
- Rover is fully on 5° slope and continues purely forward.
- Rover is commanded to first drive sideways with $v_y^* = 0.02\,\frac{\mathrm{m}}{\mathrm{s}}$ and then additionally rotate with $\omega_z^* = -0.03\,\frac{\mathrm{rad}}{\mathrm{s}}$.
- Rover drives purely forward, round about orthogonal to the inclination.

On rover level, tracking of the trajectory by the rigid body dynamics feedback linearization controller is achieved as shown in figure 6, top plot. The solid lines, representing the measured rover velocities, match the dashed trajectories nicely in forward (blue), sideways (red) and yaw (yellow) velocity. In part (b), however, the rover hits the 5° slope with the front and middle wheel pair one after the other. The contact angles are non-zero during this period, leading to additional resistance. This is an unknown

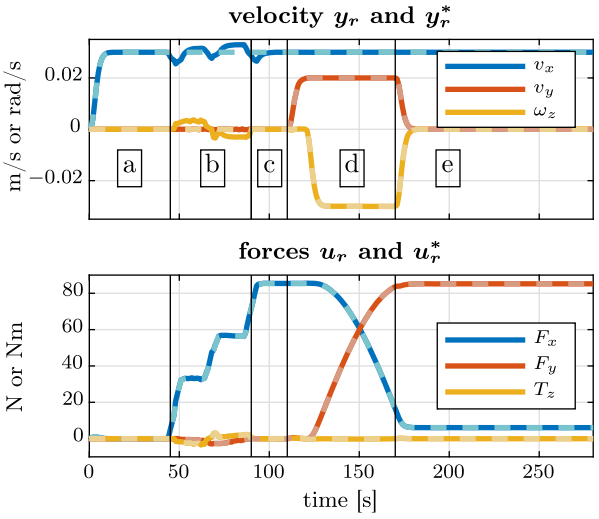

Figure 6: Rover velocity (top) and forces (bottom). Dashed, bright colored lines are the trajectory and normal, dark colored lines are the measured values. Unit and scaling of the x-axes are identical.

disturbance to the system because the contact angles were assumed zero for the controller synthesis and the downhill force is only slowly reflected in the orientation measurement. At (c) the rear wheel pair finally hits the slope and the full rover has an inclination of 5° which is measured by the attitude sensor. Due to no roughness of the ground, the contact angles, i. e., the disturbances, are back to zero. The integrator in equation (35), ensures a decent performance in part (b), despite the disturbance. Note that the yaw velocity shows some deterioration in (b) due to the rover not being perfectly aligned with the start of the slope, i. e., one wheel of the respective wheel pairs hits the slope a bit earlier than the other side.

In the bottom plot of figure 6, the trajectory (dashed) and actual (solid) total rover forces are shown. It can be seen that the gravitational forces from (b) on are much higher than the acceleration force that is needed in the beginning of (a) to start driving. Observe that in (e) almost pure y-force is needed to drive, although the rover is commanded to drive in its x-direction. This is again due to the gravitational force that is acting mostly in lateral rover direction in (e). The allocation and underlying wheel controllers achieve a good match of the measured total rover forces with the trajectory, see figure 6.

To analyze the allocation and wheel controllers in more detail, four of the six wheel forces are plotted in figure 7. The blue plots show the x- and the red plots the y-direction of the wheels in the non-steered s-frame. Looking at the bold lines – dashed is again the trajectory, solid is the actual force – the underlying wheel-ground controllers show no significant error in the whole scenario.

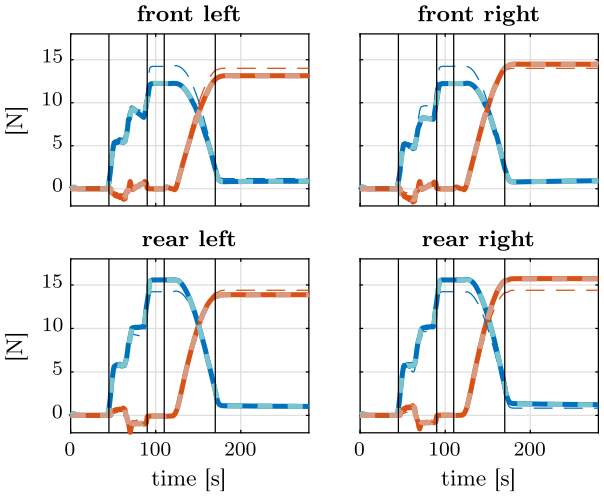

Figure 7: Wheel forces of the corner wheels, blue is x-direction, red is y in the non-steered s-frame. Bold dashed lines are the trajectories, thin dashed lines the trajectories without optimization ($\boldsymbol{\rho}^* = \mathbf{0}$) and the solid lines represent the measured values. Axes units and scaling is identical for all subplots.

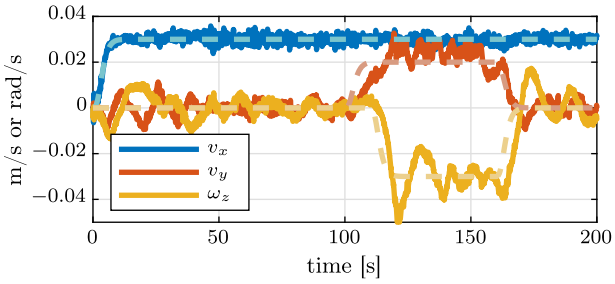

Figure 8: Rover velocity trajectory following for the large disturbances of a high-fidelity contact model with soil deformation and geometrically uneven soil. Bright, dashed lines are the trajectories and dark solid lines the actually achieved velocities. This result corresponds to the shown 3D animation from figure 5.

The optimized allocation can be analyzed by comparing the bold dashed lines (optimized allocation) with the thin dashed lines (non-optimized allocation, i. e., $\boldsymbol{\rho}^* = \mathbf{0}$). Before hitting the inclination, the least-squares solution is optimal but with more and more inclination of the rover, the normal contact forces shift to the downhill wheels. In (c), these are the rear wheels, thus the optimization algorithm computes non-zero values for $\boldsymbol{\rho}^*$ that cause higher traction forces at the rear wheels. Within part (d), the rover turns right, the right wheels are more and more downhill and thus get allocated more traction force than the left ones.

In conclusion, a good share of the rover force trajectory is handled by the model-based feedback linearization controller, e. g., based on the compensation of gravitational forces, instead of fully relying on the integrator part of a PID controller. The optimization-based control allocation allows a better utilization of the traction force potential for varying normal forces, than would be achieved with a purely velocity-based kinematic controller.

A first robustness test uses the non-nominal DLR soil contact model (SCM) [7], which is based on a 2-dimensional mesh of the ground and includes algorithms for soil deformation and soil flow underneath the wheels, see figure 5. Additionally, as can be seen in the same figure, the ground is not flat, but has a noise-based geometric unevenness. These two non-nominal conditions lead to considerable disturbance in the zero contact angle assumption and the wheel-ground force model. The resulting rover velocity trajectories for this robustness test are shown in figure 8, were the impact of the disturbances can clearly be seen. High-frequency deviations from the trajectory can be observed, which stem from the contact force computation of SCM, more specifically from the effects of the grousers entering and exiting the soil as well as the soil discretization. The lower frequency oscillations in the lateral and yaw velocity v_y and ω_z can be explained with a combination of differences in the force computation, discretization effects and, on the controller side, the lacking knowledge of the steering actuator torque limits.

More analysis of the performance in comparison to state of the art controllers and robustness tests are planned, including hardware tests with the ExoMars BB2 rover in our DLR-laboratory.

5 Conclusion

This contribution started with placing the chassis control system within the overall controls hierarchy for a wheeled mobile robot and the presentation of existing work in this field. The modular modeling approach of the mobile robot, consisting of a rigid body dynamics model, a (differential) kinematics model and a combined wheel and ground contact model, was presented. Model modifications were made to ensure controllability of the subsystems and feasibility of the allocation. A feedback linearization controller was designed for the rigid body dynamics and the underlying wheel-ground dynamics, while the control allocation was done with an approach based on the pseudo-inverse and a base of the null-space. For the optimization of the degrees of freedom, a simple objective function was chosen that keeps a maximum distance from soil failure. The performance of the overall control system and each of its com-

ponents was investigated in a software-in-the-loop simulation with a high fidelity model of the ExoMars rover.

Future work should be dedicated to an extension of the used ground contact model, testing the controller on the ExoMars BB2 rover and the implementation of contact angles into the approach for very rough terrain. It is also planned to investigate the performance for different ground types to ultimately get to an adaptive control system for heterogeneous ground.

References

1. Hamzah M Beakawi Al-Hashemi and Omar S Baghabra Al-Amoudi. A review on the angle of repose of granular materials. *Powder Technology*, 330:397–417, 2018.
2. Raymond E Arvidson, JF Bell, P Bellutta, NA Cabrol, JG Catalano, J Cohen, LS Crumpler, DJ Des Marais, TA Estlin, WH Farrand, et al. Spirit mars rover mission: Overview and selected results from the northern home plate winter haven to the side of scamander crater. *Journal of Geophysical Research: Planets*, 115(E7), 2010.
3. Raymond E Arvidson, Karl D Iagnemma, Mark Maimone, Abigail A Fraeman, Feng Zhou, Matthew C Heverly, Paolo Bellutta, David Rubin, Nathan T Stein, John P Grotzinger, et al. Mars science laboratory curiosity rover megaripple crossings up to sol 710 in gale crater. *Journal of Field Robotics*, 34(3):495–518, 2016.
4. Stefan Barthelmes. Terra: Terramechanics for real-time application. In *5th Joint International Conference on Multibody System Dynamics*, 06 2018. URL https://elib.dlr.de/121815/.
5. Stefan Barthelmes and Sebastian Zehnter. An all-terrain-controller for over-actuated wheeled mobile robots with feedforward and optimization-based control allocation. In *Proceedings of the 56th IEEE Conference on Decision and Control (CDC)*, pages 5215–5222. IEEE, 2017.
6. Jonathan Brembeck and Peter Ritzer. Energy optimal control of an over actuated robotic electric vehicle using enhanced control allocation approaches. In *Proceedings of the Intelligent Vehicles Symposium (IV)*, pages 322–327. IEEE, 2012.
7. Fabian Buse. Using superposition of local soil flow fields to improve soil deformation in the DLR soil contact model – SCM. In *Proceedings of the 5th Joint International Conference on Multibody System Dynamics (IMSD)*, 2018.
8. Luca Caracciolo, Alessandro De Luca and Stefano Iannitti. Trajectory tracking control of a four-wheel differentially driven mobile robot. In *Proceedings of the International Conference on Robotics and Automation (ICRA)*, volume 4, pages 2632–2638. IEEE, 1999.
9. Liang Ding, Zongquan Deng, Haibo Gao, Keiji Nagatani and Kazuya Yoshida. Planetary rovers' wheel – soil interaction mechanics: new challenges and applications for wheeled mobile robots. *Intelligent Service Robotics*, 4(1):17–38, 2011.
10. Alex Ellery. *Planetary Rovers – Robotic Exploration of the Solar System*. Springer, 2016. ISBN 9783642032585.
11. Ramon Gonzalez and Karl Iagnemma. Soil embedding avoidance for planetary exploration rovers. In *Proceedings of the International Conference of the International Society for Terrain-Vehicle Systems (ISTVS)*, 2016.
12. Matthias Hellerer, Tobias Bellmann and Florian Schlegel. The DLR visualization library-recent development and applications. In *Proceedings of the 10th International Modelica Conference*; March 10–12; 2014; Lund; Sweden, number 96, pages 899–911. Linköping University Electronic Press, 2014.
13. Matthias Hellerer, Stefan Barthelmes and Fabian Buse. The DLR rover simulation toolkit. In *Proceedings of the 14th Symposium on Advanced Space Technologies in Robotics and Automation (ASTRA)*, 2017.
14. Daniel M Helmick, Yang Cheng, Daniel S Clouse, Larry H Matthies and Stergios I Roumeliotis. Path following using visual odometry for a mars rover in high-slip environments. In *Proceedings of the Aerospace Conference*, volume 2, pages 772–789. IEEE, 2004.
15. Jens Hoedt. *Fahrdynamikregelung für fehlertolerante X-by-Wire-Antriebstopologien*. PhD thesis, Technische Universität Darmstadt, 2013.
16. Karl Iagnemma and Steven Dubowsky. *Mobile Robots in Rough Terrain: Estimation, Motion Planning, and Control with Application to Planetary Rovers*. Springer, 2004. ISBN 9783540219682.
17. Karl Iagnemma and Steven Dubowsky. Traction control of wheeled robotic vehicles in rough terrain with application to planetary rovers. *The International Journal of Robotics Research*, 23 (10-11):1029–1040, 2004.
18. Genya Ishigami, Keiji Nagatani and Kazuya Yoshida. Path planning for planetary exploration rovers and its evaluation based on wheel slip dynamics. In *Proceedings of the International Conference on Robotics and Automation (ICRA)*, pages 2361–2366. IEEE, 2007.
19. Alberto Isidori. *Nonlinear Control Systems*. Springer, 3 edition, 1995. ISBN 3540199160.
20. Tor A Johansen and Thor I Fossen. Control allocation – a survey. *Automatica*, 49(5):1087–1103, 2013.
21. Hassan K. Khalil. *Nonlinear Systems*. Prentice Hall, 3 edition, 2002.
22. Krzysztof Kozłowski and Dariusz Pazderski. Modeling and control of a 4-wheel skid-steering mobile robot. *International Journal of Applied Mathematics and Computer Science*, 14:477–496, 2004.
23. Ambroise Krebs, Thomas Thueer, Edgar Carrasco, Raymond Oung and Roland Siegwart. Towards torque control of the crab rover. In *Proceedings of the 9th International Symposium on Artificial Intelligence, Robotics and Automation in Space (iSAIRAS'08)*, 2008.
24. Rainer Krenn, Andreas Gibbesch, Giovanni Binet and Alberto Bemporad. Model predictive traction and steering control of planetary rovers. In *Proceedings of the Symposium on Advanced Space Technologies in Robotics and Automation (ASTRA)*, 2013.
25. Rainer Krenn, Johannes Köppern, Tilman Bünte, Jonathan Brembeck, Andreas Gibbesch and Johann Bals. Modellbasierte regelungsansätze für überaktuierte planetare rover und robotische elektromobile. *at-Automatisierungstechnik*, 61(3): 183–194, 2013.
26. Pierre Lamon and Roland Siegwart. Wheel torque control in rough terrain-modeling and simulation. In *Proceedings of the International Conference on Robotics and Automation (ICRA)*,

pages 867–872. IEEE, 2005.
27. Damien Lhomme-Desages, Christophe Grand and Jean-Claude Guinot. Trajectory control of a four-wheel skid-steering vehicle over soft terrain using a physical interaction model. In *Proceedings of the International Conference on Robotics and Automation (ICRA)*, pages 1164–1169. IEEE, 2007.
28. Vivian Medeiros, Diego Rosa and Marco Meggiolaro. Torque optimization for stability control of wheeled vehicles in rough terrain. In *Proceedings of the 18th International Symposium on Dynamic Problems of Mechanics (DINAME)*, 2019.
29. Stéphane Michaud, Andreas Gibbesch, Thomas Thüer, Ambroise Krebs, Christopher Lee, B Despont, Bernd Schäfer, Richard Slade. Development of the exomars chassis and locomotion subsystem. In *Proceedings of the 9th International Symposium on Artificial Intelligence, Robotics and Automation in Space (i-SAIRAS)*. Eidgenössische Technische Hochschule Zürich, Autonomous Systems Lab, 2008.
30. Jan-Erik Moseberg. *Regelung der Horizontalbewegung eines überaktuierten Fahrzeugs unter Berücksichtigung von Realisierungsanforderungen*. PhD thesis, Friedrich-Alexander-Universität Erlangen-Nürnberg, 2016.
31. Ralf Orend. Modelling and control of a vehicle with single-wheel chassis actuators. In *Proceedings of the IFAC World Congress*, volume 38, pages 79–84. Elsevier, 2005.
32. Thierry Peynot and Simon Lacroix. Enhanced locomotion control for a planetary rover. In *Proceedings on the International Conference on Intelligent Robots and Systems (IROS)*, volume 1, pages 311–316. IEEE, 2003.
33. Bruno Siciliano, Lorenzo Sciavicco, Luigi Villani and Giuseppe Oriolo. *Robotics: Modelling, Planning and Control*. Springer, 2010. ISBN 9781846286414.
34. Alexandre F Barral Silva, Auderi Vicente Santos, Marco Antonio Meggiolaro and Mauro Speranza Neto. A rough terrain traction control technique for all-wheel-drive mobile robots. *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, 32(4):489–501, 2010.
35. Benoit Thuilot, B d'Aandrea Novel and Alain Micaelli. Modeling and feedback control of mobile robots equipped with several steering wheels. *IEEE Transactions on Robotics and Automation*, 12 (3):375–390, 1996.
36. Olivier Toupet, Jeffrey Biesiadecki, Arturo Rankin, Amanda Steffy, Gareth Meirion-Griffith, Dan Levine, Maximilian Schadegg and Mark Maimone. Traction control design and integration onboard the mars science laboratory curiosity rover. In *Proceedings of the Aerospace Conference*, pages 1–20. IEEE, 2018.
37. Kenneth J Waldron and Muhammad E Abdallah. An optimal traction control scheme for off-road operation of robotic vehicles. *IEEE/ASME Transactions on Mechatronics*, 12 (2):126–133, 2007.
38. Jo Yung Wong. *Theory of Ground Vehicles*. John Wiley & Sons, Inc., Hoboken, New Jersey, USA, 4 edition, 2008. ISBN 9780471354611.
39. Katarzyna Zadarnowska and Adam Oleksy. Motion planning of wheeled mobile robots subject to slipping. *Journal of Automation Mobile Robotics and Intelligent Systems*, 5:49–58, 2011.

Bionotes

Stefan Barthelmes
Deutsches Zentrum für Luft- und Raumfahrt e.V., Institut für Systemdynamik und Regelungstechnik, Münchener Str. 20, 82234 Weßling, Germany
stefan.barthelmes@dlr.de

Stefan Barthelmes is a research associate in the planetary exploration group of the DLR Institute of System Dynamics and Control. He holds Bachelor's and Master's degrees in Mechanical Engineering from Technische Universität München and is currently pursuing a PhD on chassis controls for planetary exploration rovers with Prof. Konigorski. Besides controls engineering for mobile robots, he is working on the multi-physics modeling of robots as well as real-time capable terramechanics models.

Ulrich Konigorski
TU Darmstadt, Institut für Automatisierungstechnik und Mechatronik, Fachgebiet Regelungstechnik und Mechatronik, Landgraf-Georg-Str. 4, 64283 Darmstadt, Germany
ukonigorski@iat.tu-darmstadt.de

Prof. Dr.-Ing. Ulrich Konigorski is head of the Control Systems and Mechatronics Laboratory at the Department of Electrical Engineering and Information Technology at Technische Universität Darmstadt. His methodical focus is on the linear and nonlinear design of multiple-input multiple-output systems, multi-agent systems and modeling and design of distributed systems. He mainly applies these methods to the modeling, design and digital control of mechatronic systems.