DINAME 2019 - Proceedings of the XVIII International Symposium on Dynamic Problems of Mechanics,
M.A. Savi, T.G. Ritto and W.M. Bessa (Editors), ABCM, Buzios, RJ, Brazil, March 10th to 15th, 2019

Control of Multibody Systems in DAE Form Applied to a Cable Robot

Svenja Otto[1], **Tobias Rückwald**[1], **and Robert Seifried**[1]

[1] Hamburg University of Technology, Mechanics and Ocean Engineering, Eißendorfer Str. 42, 21073 Hamburg, Germany

Abstract: Certain types of multibody systems cannot be conveniently modelled using minimal coordinates. Then, redundant coordinates can be chosen to represent the system kinematics. This results in the equations of motion being described by differential-algebraic equations (DAEs). Control of systems described by DAEs is not straight-forward in practical applications. In this paper, application and implementation of a linear quadratic regulator is shown for underactuated systems in DAE form. Therefore, the equations of motion are linearised around the desired state trajectory and transformed into ordinary differential equations using Maggi's formulation with application of the QR decomposition. For implementation of the state feedback controller, measurements or estimates of the full state vector are required. For observer design, an unscented Kalman filter is proposed which is able to account for measurements of the algebraic coordinates. Application of the controller in a two-degree of freedom control structure is shown in simulations and experiments on an underactuated cable robot.

Keywords: Multibody Systems, DAEs, Feedback Control, Cable Robot, UKF

INTRODUCTION

Controller design for multibody systems requires a sufficiently accurate model of the dynamic behaviour. Most frequently, multibody systems are modelled using generalized coordinates which specify a minimal number of coordinates that describe the system kinematics. The resulting dynamic equations are ordinary differential equations (ODEs). A simple and standard controller design for linear systems modelled as ODEs is a linear quadratic regulator. Sometimes, for example in case the multibody system contains kinematic loops, it is not possible to conveniently describe the dynamics using minimal coordinates. Then, redundant coordinates are chosen which result in differential-algebraic equations (DAEs) describing the dynamic behaviour. Controller design for DAE systems is not straight-forward in practical applications. This is especially true if these systems are also underactuated, meaning they possess less system inputs than degrees of freedom. However, underactuation is quite common and can arise due to different reasons, such as body elasticities or intentional design specifications. Control of underactuated systems by means of feedback linearisation is e.g. presented by Seifried (2014).

There are several approaches for control of DAE systems. They usually extend classical control design methods for systems described by DAEs. For example, the extension of linear model-predictive control is described by Findeisen and Allgöwer (2000) for index 1 DAEs. The extension of optimal control and a linear quadratic regulator is shown for example by Heiland (2016) for flow control problems.

The previously mentioned methods are state feedback controllers. However, it is usually not possible to measure all redundant coordinates directly. In that case, an observer can be applied to estimate unmeasurable states, see e.g. Thrun et al. (2006). For systems described by linear ordinary differential equations, the Kalman filter is a standard observer. The estimation process consists of two parts. At first, the model equations are evaluated in the prediction step. Afterwards, measurement data is used to improve the prediction in the update step. For nonlinear systems, the Kalman filter was extended to an unscented Kalman filter (UKF) by Julier and Uhlmann (2004). Thereby, the nonlinearities in the state equations are estimated by the unscented transform. There are several approaches to extend the UKF for systems described by DAEs, see e.g. the survey by Patwardhan et al. (2012). Mandela et al. (2010) note that DAE based observer design is more complex due to two reasons. First, some approaches do not allow for measurements of the algebraic coordinates. Second, the algebraic constraints are not necessarily fulfilled after inclusion of the measurement data in the update step. This yields initialization problems in the next time step. In this case, a reevaluation of the algebraic constraints after the update step is proposed in order to ensure a consistent prediction, see Mandela et al. (2010). However, this approach is limited to index 1 DAEs.

In this paper, application and implementation of a linear quadratic regulator for underactuated mechanical descriptor systems is shown. Moreover, an UKF is proposed for DAE systems to observe unmeasureable states, which fulfils the equations of motion on velocity level and which includes measurements of algebraic states. Moreover, experiments are performed on an underactuated cable robot to support the results and an evaluation of computational efficiency is performed.

MODELLING

A holonomic multibody system with n degrees of freedom and m system inputs is considered. It is assumed that the system is underactuated, i.e. $m < n$. The system is modelled using redundant position coordinates $\mathbf{y} \in \mathbb{R}^{n+n_c}$. Thereby, n_c denotes the number of algebraic equations constraining the motion of the system. The equations of motion arise in the form

$$\mathbf{M}\ddot{\mathbf{y}} = \mathbf{q}(\mathbf{y},\dot{\mathbf{y}},t) + \mathbf{C}^\top(\mathbf{y},t)\boldsymbol{\lambda} + \mathbf{B}\mathbf{u} \tag{1}$$

$$\mathbf{c}(\mathbf{y},t) = \mathbf{0}, \tag{2}$$

with the mass matrix $\mathbf{M}$ and vector of generalized forces $\mathbf{q}(\mathbf{y},\dot{\mathbf{y}},t)$. The geometric constraints $\mathbf{c}(\mathbf{y},t)$ are enforced by Lagrange multipliers $\boldsymbol{\lambda} \in \mathbb{R}^{n_c}$, which are mapped onto the directions of the coordinates by the Jacobian matrix

$$\mathbf{C}(\mathbf{y},t) = \frac{\partial \mathbf{c}(\mathbf{y},t)}{\partial \mathbf{y}}. \tag{3}$$

For general multibody systems, the Lagrangian multipliers are generalized reaction forces. The system inputs $\mathbf{u} \in \mathbb{R}^m$ are distributed by the matrix $\mathbf{B}$. The system output $\mathbf{z} \in \mathbb{R}^m$ is defined as a function of the position coordinates, such that

$$\mathbf{z} = \mathbf{h}(\mathbf{y}). \tag{4}$$

DAEs can be classified based on the differentiation index, see Campbell and Gear (1995). Loosely speaking, the index is defined as the number of derivatives of the algebraic constraints $\mathbf{c}(\mathbf{y},t)$ necessary for the set of Eqs. (1)-(2) to be transformed to ODEs. For multibody systems described by Eqs. (1)-(2), the differentiation index is usually 3.

Linearisation

Using Taylor expansion, the nonlinear Eqs. (1)-(2) are linearised around the desired state trajectories $\mathbf{y}_\mathrm{d}(t)$, $\dot{\mathbf{y}}_\mathrm{d}(t)$ and $\boldsymbol{\lambda}_\mathrm{d}(t)$ of the coordinates $\mathbf{y}$, $\dot{\mathbf{y}}$ and $\boldsymbol{\lambda}$ respectively. Obtaining the desired state trajectories from the desired output trajectory $\mathbf{z}_\mathrm{d}(t)$ is not straight-forward for the case of underactuated multibody systems. An inverse model is necessary for that mapping and details are discussed below in the control section. Assuming the state trajectories are available, the linearisation yields a linear time variant DAE system of the form

$$\mathbf{E}\dot{\tilde{\mathbf{x}}} = \mathbf{A}(t)\,\tilde{\mathbf{x}} + \mathbf{B}(t)\,\tilde{\mathbf{u}} \tag{5}$$

with time dependent coefficient matrices $\mathbf{A}(t)$ and $\mathbf{B}(t)$, a singular matrix $\mathbf{E}$ with rank $2(n+n_c)$ and the state vector $\tilde{\mathbf{x}} = \begin{bmatrix}\tilde{\mathbf{y}} & \dot{\tilde{\mathbf{y}}} & \tilde{\boldsymbol{\lambda}}\end{bmatrix}^\top$ that denotes small variations with respect to the desired state trajectories $\mathbf{y}_\mathrm{d}(t)$, $\dot{\mathbf{y}}_\mathrm{d}(t)$ and $\boldsymbol{\lambda}_\mathrm{d}(t)$. Choosing a stationary reference point, the time variant matrices $\mathbf{A}(t)$ and $\mathbf{B}(t)$ reduce to constant matrices $\mathbf{A}$ and $\mathbf{B}$.

Model Transformation

The most straight-forward method of transforming DAEs into ODEs is to differentiate the constraints $\mathbf{c}(\mathbf{y},t)$. For general multibody systems, this involves two differentiations and some algebraic transformations for elimination of the Lagrangian multipliers $\boldsymbol{\lambda}$. Therefore, the constraints are only fulfilled on acceleration level. During the integration process, this yields a quadratic numerical drift. In order to significantly reduce numerical drift, the DAE model is transformed to an ODE applying Maggi's approach, e.g. summarized by Bauchau (2011). The transformation uses a projection matrix $\mathbf{J} \in \mathbb{R}^{(n+n_c)\times n}$ such that the generalized reaction forces $\mathbf{C}^\top(\mathbf{y},t)\boldsymbol{\lambda}$ vanish. The projection is based on a QR decomposition of the Jacobian matrix of the constraints

$$\mathbf{C}^\top = \mathbf{Q}\mathbf{R} = \begin{bmatrix}\mathbf{Q}_1 & \mathbf{Q}_2\end{bmatrix}\begin{bmatrix}\mathbf{R}_1 \\ \mathbf{0}\end{bmatrix}, \tag{6}$$

as proposed by Kim and Vanderploeg (1986). The columns of the matrix $\mathbf{Q}_1$ represent a orthonormal basis of the columnspace of $\mathbf{C}^\top$. The time derivatives of the constraint $\mathbf{c}(\mathbf{y})$ are computed with

$$\dot{\mathbf{c}} = \mathbf{C}\dot{\mathbf{y}} + \frac{\partial \mathbf{c}}{\partial t} = \mathbf{C}\dot{\mathbf{y}} + \mathbf{c}_t \tag{7}$$

$$\ddot{\mathbf{c}} = \mathbf{C}\ddot{\mathbf{y}} + \dot{\mathbf{C}}\dot{\mathbf{y}} + \dot{\mathbf{c}}_t = \mathbf{C}\ddot{\mathbf{y}} + \mathbf{c}_{tt}. \tag{8}$$

With the abbreviations $\boldsymbol{\beta} = -\mathbf{Q}_1\mathbf{R}_1^{-\top}\mathbf{c}_t$ and $\boldsymbol{\gamma} = -\mathbf{Q}_1\mathbf{R}_1^{-\top}\mathbf{c}_{tt}$ and choosing the projection as $\mathbf{J} = \mathbf{Q}_2$, the ODE representation of the dynamic Eqs. (1)-(2) is given by

$$\begin{bmatrix}\dot{\mathbf{y}} \\ \ddot{\mathbf{y}}\end{bmatrix} = \begin{bmatrix}\mathbf{J}\mathbf{J}^\top\dot{\mathbf{y}} + \boldsymbol{\beta} \\ \mathbf{J}\left(\left(\mathbf{J}^\top\mathbf{M}\mathbf{J}\right)^{-1}\left(\mathbf{J}^\top\mathbf{q} + \mathbf{J}^\top\mathbf{B}\mathbf{u} - \mathbf{J}^\top\mathbf{M}\boldsymbol{\gamma}\right)\right) + \boldsymbol{\gamma}\end{bmatrix}. \tag{9}$$

The ordinary differential Eqs. (9) fulfil the constraints on velocity level and are of dimension $2(n+n_c)$. Further stabilization is possible by applying Baumgarte stabilization to the constraint Eq. (2). Note that in using Eqs. (9), no coordinate transformation is necessary, since the ODEs are formulated in the original redundant coordinates. In the following, Maggi's approach is directly applicable to the nonlinear Eqs. (1)-(2), but also to the linearised version.

CONTROLLER DESIGN

The chosen control design approach is a two-degree of freedom control structure. A feedforward control part is responsible for trajectory tracking, while the state feedback controller stabilizes the system around the desired trajectory.

Feedforward Control

An inverse model serves as feedforward control. In order to account for underactuated multibody systems, the servo-constraints approach is applied here. The method is directly applicable to systems written in DAE from, see e.g. Otto and Seifried (2018). The equations of motion (1) and (2) are extended to include servo-constraints

$$\mathbf{0} = \mathbf{z}(\mathbf{y}) - \mathbf{z}_\mathrm{d}(t) \tag{10}$$

that constrain the system output $\mathbf{z}$ to follow the specified trajectory $\mathbf{z}_\mathrm{d}(t)$. Combining Eqs. (1)-(2) and (10) yields a new set of DAEs. These DAEs describe the inverse model of the system and might be of higher differentiation index or contain unbounded internal dynamics, see Seifried (2014). They are solved in each time step for the desired state trajectories $\mathbf{y}_\mathrm{d}(t)$, $\dot{\mathbf{y}}_\mathrm{d}(t)$ and $\boldsymbol{\lambda}_\mathrm{d}(t)$ and the desired control action $\mathbf{u}_\mathrm{d}$ that exactly moves the system on the specified trajectory $\mathbf{z}_\mathrm{d}(t)$ in case of no disturbances or modelling errors.

Feedback Control

For state feedback control, two approaches are compared here. The first approach is proposed by Bender and Laub (1987). Based on the linearised DAE form of Eqs. (5), a Riccati algebraic equation is directly formulated for the DAE and can be solved for the linear time invariant case. The second approach is based on the nonlinear Eqs. (1)-(2), which are linearised and projected by Maggi's formulation with QR decomposition to obtain the ordinary differential Eqs. (9) in linear form. Then, standard linear quadratic control is applied based on the linear projected equations. Thereby, the linearisation can be performed with respect to a time dependent desired trajectory $\mathbf{z}_\mathrm{d}(t)$ or a constant reference point, yielding a linear time variant (LTV) or linear time invariant (LTI) system, respectively. The optimal state feedback control input for both formulations is

$$\mathbf{u} = \mathbf{K}(t)\left(\mathbf{x} - \mathbf{x}_\mathrm{d}(t)\right), \tag{11}$$

with the controller gain matrix $\mathbf{K}(t) \in \mathbb{R}^{m\times(2(n+n_c))}$ and state vector $\mathbf{x} = \begin{bmatrix}\mathbf{y} & \dot{\mathbf{y}}\end{bmatrix}^\top$. In the LTI case, the gain matrix reduces to a constant matrix $\mathbf{K}$. The cost function for the optimal controller gain is defined as

$$J = \frac{1}{2}\int_0^\infty \mathbf{x}^\top \mathbf{Q}_\mathrm{LQR}\,\mathbf{x} + \mathbf{u}^\top \mathbf{R}_\mathrm{LQR}\,\mathbf{u}\;\mathrm{d}t \tag{12}$$

with the weighting matrices $\mathbf{Q}_\mathrm{LQR} \in \mathbb{R}^{(2(n+n_c))\times(2(n+n_c))}$ and $\mathbf{R}_\mathrm{LQR} \in \mathbb{R}^{m\times m}$. The solution of the optimization problem is obtained by solving the algebraic Riccati equation or the differential Riccati equations for the LTI or LTV case respectively.

Note that the ordinary differential Eqs. (9) obtained from Maggi's approach are written in the original redundant coordinates. Therefore, the weighting matrices $\mathbf{Q}_\mathrm{LQR}$ and $\mathbf{R}_\mathrm{LQR}$ can be tuned intuitively with the physical coordinates in mind. This is also true for the approach from Bender and Laub (1987), who apply Riccati's algebraic equation directly on the DAE formulation. For state feedback, all redundant coordinates $\mathbf{y}$ and $\dot{\mathbf{y}}$ must be fed back. Therefore, they must either be measured or observed by a suitable DAE based observer.

OBSERVER DESIGN

The Kalman filter is an observer for LTV and LTI systems described by linear ordinary differential equations. The ODEs can be expressed in form of Eqs. (5) when choosing the matrix $\mathbf{E}$ as identity, i.e. $\mathbf{E} = \mathbf{I}$. It is assumed that the estimated values, process noise and measurement noise are distributed in a Gaussian distribution. A short overview of the filter equations is given according to Thrun et al. (2006). The Kalman filter yields estimates for the mean value $\hat{\mathbf{x}}_{k|+}$ and the covariance $\boldsymbol{\Sigma}_{k|+}$ at time instant k. At the next time instant $k+1$, the current prediction is fed through the linear dynamic equations

$$\hat{\mathbf{x}}_{k+1|-} = \mathbf{A}_{k+1}\,\hat{\mathbf{x}}_{k|+} + \mathbf{B}_{k+1}\,\mathbf{u}_{k+1} \tag{13}$$

$$\boldsymbol{\Sigma}_{k+1|-} = \mathbf{A}_{k+1}\,\boldsymbol{\Sigma}_{k|+}\,\mathbf{A}_{k+1}^\top + \mathbf{Q}_{\mathrm{UKF},k+1} \tag{14}$$

to obtain the predictions $\hat{\mathbf{x}}_{k+1|-}$ and $\boldsymbol{\Sigma}_{k+1|-}$. Thereby, the process noise is represented by the covariance matrix $\mathbf{Q}_\mathrm{UKF}$. This is called the prediction step. Afterwards, the prediction is corrected by using measurement data. Thereby, the current mean of the state vector is passed through the linear measurement function $\mathbf{z} = \mathbf{H}\mathbf{x}$ to estimate the current measurements $\hat{\mathbf{z}}_{k+1}$. Comparing the estimated measurements with the actual measurements yields an error $(\mathbf{z}_{k+1} - \hat{\mathbf{z}}_{k+1})$, which is fed back

through the Kalman gain matrix

$$\mathbf{K}_{k+1} = \boldsymbol{\Sigma}_{k+1|-} \mathbf{H}_{k+1}^{\top} \left(\mathbf{H}_{k+1} \boldsymbol{\Sigma}_{k+1|-} \mathbf{H}_{k+1}^{\top} + \mathbf{R}_{\mathrm{UKF},k+1} \right)^{-1} . \tag{15}$$

Thereby, the measurement noise is represented by its covariance matrix $\mathbf{R}_{\mathrm{UKF}}$. Then, the corrected mean and covariance values are obtained from

$$\hat{\mathbf{x}}_{k+1|+} = \hat{\mathbf{x}}_{k+1|-} + \mathbf{K}_{k+1} \left(\mathbf{z}_{k+1} - \hat{\mathbf{z}}_{k+1} \right) \tag{16}$$

$$\boldsymbol{\Sigma}_{k+1|+} = \left(\mathbf{I} - \mathbf{K}_{k+1} \mathbf{H}_{k+1} \right) \boldsymbol{\Sigma}_{k+1|-} . \tag{17}$$

This is called the correction step and completes the Kalman filter algorithm for one time step. One possible nonlinear extension is the unscented Kalman filter with the unscented transform stochastically linearising the nonlinear dynamics, see Julier and Uhlmann (2004).

Mandela et al. (2010) describe possible extensions of the UKF for system dynamics described by differential algebraic Eqs. (1)-(2). They identify two main challenges. The corrected states $\hat{\mathbf{x}}_{k+1|+}$ might be inconsistent with the algebraic Eq. (2) and their time integration might not be possible in the next prediction step. Moreover, incorporation of measurements of the algebraic states $\boldsymbol{\lambda}$ is not straight-forward because the algebraic states are typically eliminated in the process of transforming a DAE into an ODE. In contrast to the strategy proposed by Mandela et al. (2010), which works for index 1 DAEs, a method that fulfils the algebraic constraints on velocity level is proposed here. In order to avoid the problem of inconsistent states after the update step, the DAE model is again reformulated into an ODE model expressed by Eqs. (9). Then, the ordinary differential Eqs. (9) are utilized in the prediction step. For incorporation of measurements of the algebraic coordinates $\boldsymbol{\lambda}$ in the correction step, they are evaluated with help of the QR decomposition described above. Multiplying Eq. (1) from left with the matrix $\mathbf{R}_1^{-1} \mathbf{Q}_1^{\top}$ yields

$$\mathbf{R}_1^{-1} \mathbf{Q}_1^{\top} \mathbf{C}^{\top} \boldsymbol{\lambda} = \mathbf{R}_1^{-1} \mathbf{Q}_1^{\top} \left(\mathbf{M} \ddot{\mathbf{y}} - \mathbf{q} - \mathbf{B} \mathbf{u} \right) , \tag{18}$$

where the matrix $\mathbf{R}_1^{-1} \mathbf{Q}_1^{\top} \mathbf{C}^{\top}$ reduces to the identity matrix due to definition of the QR decomposition and Eq. (6). Then, the reaction forces are evaluated as

$$\boldsymbol{\lambda} = \mathbf{R}_1^{-1} \mathbf{Q}_1^{\top} \left(\mathbf{M} \ddot{\mathbf{y}} - \mathbf{q} - \mathbf{B} \mathbf{u} \right) , \tag{19}$$

where the second derivatives can be substituted by the dynamic equations. Thus, the algebraic constraints are fulfilled on velocity level and the numerical drift is expected to be smaller compared to an index 1 reformulation.

SIMULATIONS AND EXPERIMENTAL RESULTS

The proposed controller and observer are implemented on an underactuated cable robot. The experimental setup consists of a load platform that is hoisted by four cables from a trolley, see Fig. 1. The trolley can move on a set of parallel axes and the operating range is up to 9 m for hoisting and up to 12 m for trolley motion. The mass of the load platform is $m_{\mathrm{P}} = 18.9\,\mathrm{kg}$. All cables can be actuated separately for control of the platform orientation. A two-dimensional model of the experimental setup is depicted in Fig. 2. Thereby, two cables are considered to move separately and rotation of the platform around the y-axis is possible. The trolley position is denoted by s, the cable lengths are denoted by l_1 and l_2 respectively and the rotation angle of the platform is denoted by δ_{P}. The $m = 3$ system inputs are the setpoint velocity u_s of the trolley and the setpoint velocities u_{l1} and u_{l2} of the cables 1 and 2, respectively. The setpoint velocities model velocity-controlled actuators. Due to the kinematic loop consisting of both cables and the platform, a selection of minimal coordinates is not directly possible. Thus, redundant position coordinates are chosen as

$$\mathbf{y} = \begin{bmatrix} s & l_1 & l_2 & x_{\mathrm{P}} & z_{\mathrm{P}} & \delta_{\mathrm{P}} \end{bmatrix}^{\top} . \tag{20}$$

The system output is defined as the platform position and its orientation,

$$\mathbf{z} = \begin{bmatrix} x_{\mathrm{P}} & z_{\mathrm{P}} & \delta_{\mathrm{P}} \end{bmatrix}^{\top} . \tag{21}$$

It is not possible to directly measure the sway angle φ of the load platform. Force sensors are available to measure the tension of all cables and an inertial measurement unit is installed inside the load platform to measure its translational accelerations and rotational velocities. Note that the cable forces are reaction forces and therefore enter the dynamics equations in the algebraic coordinates $\boldsymbol{\lambda}$. Therefore, an observer with the ability to account for measurements in algebraic variables is necessary. The UKF described above is implemented to observe the state vector $\mathbf{y}$ based on these measurements.

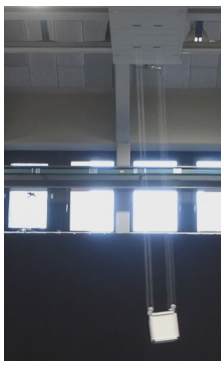

Figure 1 – Laboratory cable robot.

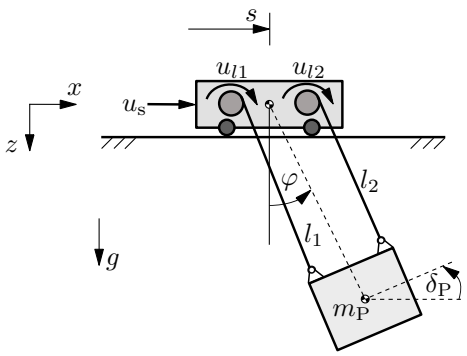

Figure 2 – Model of the experimental setup.

Control of Platform Orientation

First, control of the platform orientation is demonstrated in simulations and experiment. Thereby, the trolley position is held constant with the system input $u_s = 0\,\frac{\mathrm{m}}{\mathrm{s}}$. The control objective requires the system to be modelled in DAE form. The initial coordinates are chosen as $\mathbf{y}_0 = \begin{bmatrix}16\,\mathrm{m} & 5.35\,\mathrm{m} & 5.35\,\mathrm{m} & 16\,\mathrm{m} & 5.53\,\mathrm{m} & 0^\circ\end{bmatrix}^\top$. Then, the desired orientation angle of $\delta_{\mathrm{P,d}} = 5^\circ$ is prescribed, while the position of trolley and the position z_P are prescribed to be constant. The remaining variables x_P, l_1 and l_2 of the vector of desired coordinates are calculated to be consistent with the algebraic constraints. For experiment and simulations, the weighting matrices of the LQR controller of Eq. (11) are chosen as

$$\mathbf{Q}_{\mathrm{LQR}} = \mathrm{diag}(1.87, 7.95, 6.03, 2.58, 2.03, 0.20, 1.59, 6.15, 2.68, 1.23) \tag{22}$$

$$\mathbf{R}_{\mathrm{LQR}} = \mathrm{diag}(8000, 8000)\,. \tag{23}$$

The controller is tuned such that there is no overshoot in the platform orientation angle δ_P in order to allow a slow smooth transition of the orientation. This is necessary to avoid dangerous oscillations in the cables on the experimental setup. The simulations are performed for the controller proposed by Bender and Laub (1987) and for the one proposed in this paper, using the QR decomposition. The results are shown in Fig. 3, where the dashed lines denote controller activation. Video snapshots of the experiment are shown in Fig. 4. Looking at the simulated results in Fig. 3 shows that both controller formulations yield nearly similar results in simulations. The desired rotation angle is approached after $t = 45\,\mathrm{s}$ by actuating the cables in opposite directions, see the lower graphs of Fig. 3. The change of orientation comes along with a shift of the centre of gravity in x-direction, see the upper right hand graph. Due to similar results and more flexibility regarding LTV capabilities, the proposed controller based on Maggi's formulation is used in the experiment. Note that for the experimental results shown in Fig. 3, the orientation angle shown in the upper left hand graph is estimated by the UKF. The estimated platform orientation approaches the desired angle $\delta_\mathrm{P} = 5^\circ$ a little bit faster than in the simulation, but the general behaviour looks similar to the simulations.

Figure 5 presents the measured accelerations of the inertial measurement unit installed on the load platform. This shows that the estimated angle is not equal to the true angle. In the static equilibrium, which is reached after approximately 30 s, the measured accelerations in x- and z-direction can be utilized to calculate the true orientation by comparing the measured accelerations a_x and a_z in x- and z-direction, respectively, to the gravity vector. The true orientation $\delta_{\mathrm{P,real}}$ is approximately $\delta_{\mathrm{P,real}} = \arctan(-a_x/a_z) \approx 4^\circ$ in contrast to the estimated angle $\delta_\mathrm{P} = 5^\circ$. Thus, more tuning of the UKF design parameters $\mathbf{Q}_{\mathrm{UKF}}$ and $\mathbf{R}_{\mathrm{UKF}}$ is necessary for the observer to track the true orientation angle accurately. However, this experiment shows that the combination of the DAE based observer and the DAE based controller is stable and reaches the desired final orientation angle.

Damping of Load Platform Sway

For a second experiment, the model is reduced. Now, both cables are actuated synchronously and platform rotation is constant with $\delta_\mathrm{P} = 0^\circ$. Therefore, the vector of redundant coordinates reduces to

$$\mathbf{y} = \begin{bmatrix}s & l & x_\mathrm{P} & z_\mathrm{P}\end{bmatrix} \tag{24}$$

with the combined cable length $l = l_1 = l_2$. This model does not contain any kinematic loops and therefore the equations of motion can be written in ODE form with generalized coordinates

$$\mathbf{y}_{gen} = \begin{bmatrix}s & l & \varphi\end{bmatrix}^\top . \tag{25}$$

This model is chosen to compare the control performance of the DAE based controller to a linear quadratic controller based on the linearised ODE model. Note that for both controllers, the same ODE based UKF is used since its performance and

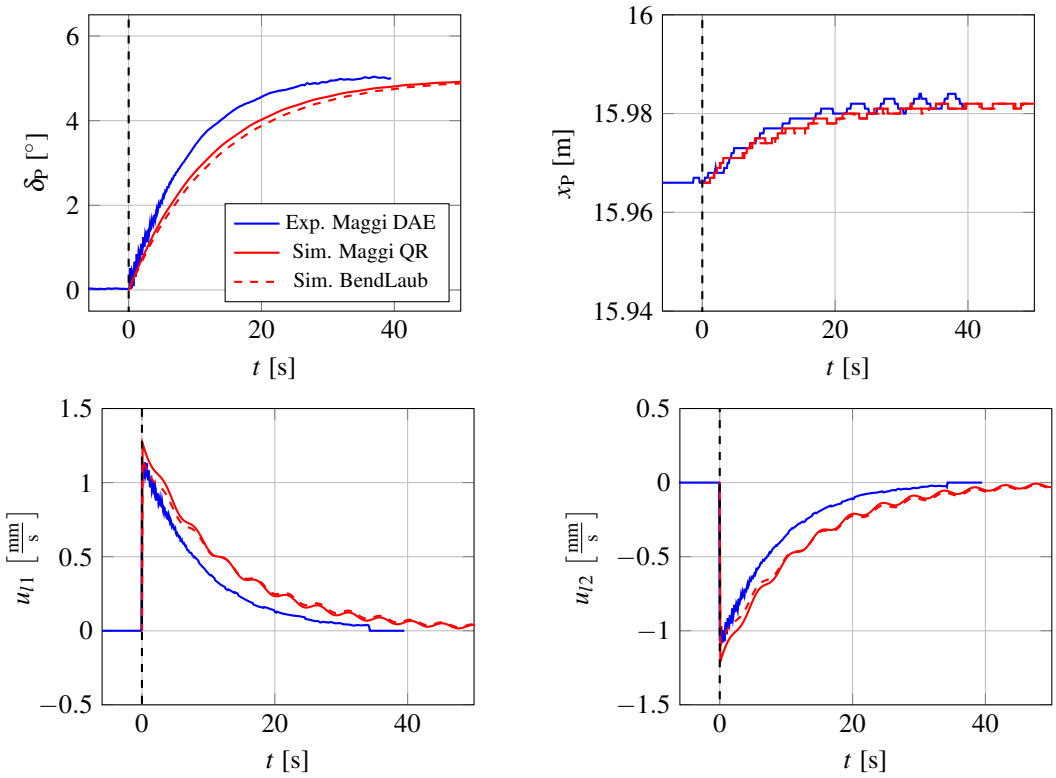

Figure 3 – Experimental and simulation results for platform orientation control.

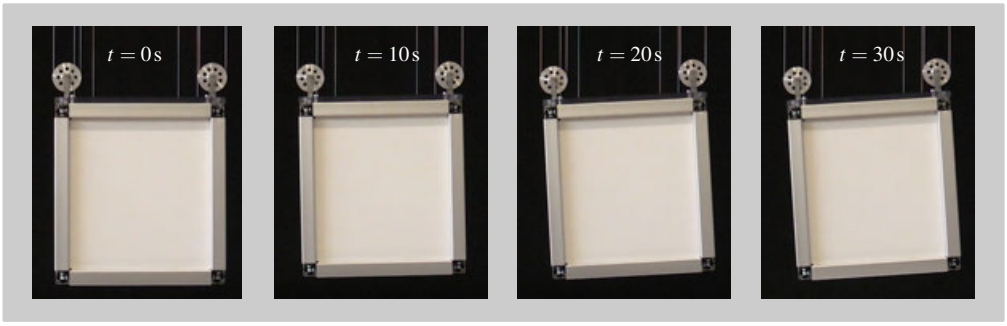

Figure 4 – Video snapshots of platform orientation control.

accurateness is verified in Kreuzer et al. (2014). For the DAE based controller, the weighting matrices are chosen as

$$\mathbf{Q}_{\mathrm{LQR}} = \mathrm{diag}(1.87, 7.95, 6.03, 2.58, 0.2, 1.6, 6.15, 2.69, 0) \tag{26}$$

$$\mathbf{R}_{\mathrm{LQR}} = \mathrm{diag}(30, 30)\,, \tag{27}$$

and for the ODE based one, they are chosen as

$$\mathbf{Q}_{\mathrm{LQR}} = \mathrm{diag}(0.0025, 2.168, 2.5274, 0.0024, 2.168, 0.0009) \tag{28}$$

$$\mathbf{R}_{\mathrm{LQR}} = \mathrm{diag}(0.019, 2.4023)\,. \tag{29}$$

The platform is excited to perform sway motion φ with maximum angle of approximately 7.3° by inducing a constant trolley velocity of $u_s = -0.5\,\frac{\mathrm{m}}{\mathrm{s}}$ for a duration of 2 s. After another 3.5 s, the controller is activated and platform sway is reduced by defining a constant desired position. The platform sway angle φ, the system input u_s of the trolley and the trolley position s are shown in Fig. 6. The excitation of the experimental setup is visible for times $t < 0\,\mathrm{s}$. Controller activation is again denoted by the dashed black line at time $t = 0\,\mathrm{s}$. The results show that a reduction of the sway angle φ with the DAE based controller is possible. The ODE based controller dampens out the sway motion more quickly compared to the DAE based version. To achieve this, it also uses more input energy, which is depicted in the upper right hand graph of Fig. 6. The total input, expressed by $\int u_s^2\,\mathrm{d}t$, of the ODE controller is approximately 3.5 times more than

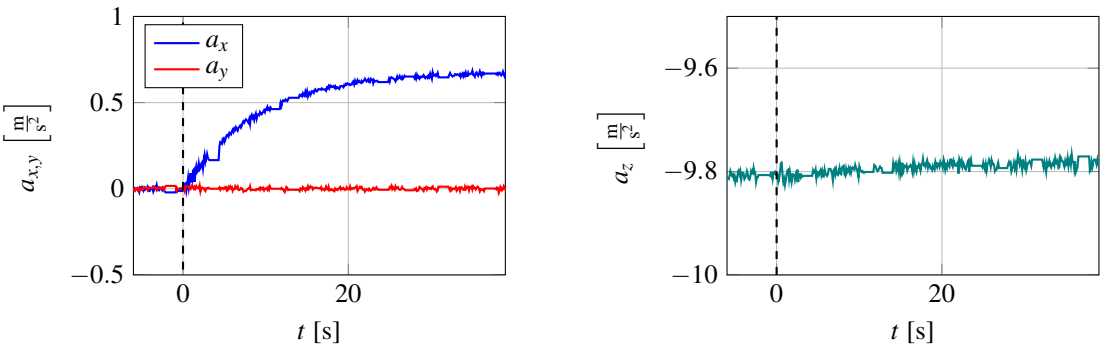

Figure 5 – Acceleration measurements for the platform orientation control experiment.

the total input of the DAE based controller. The total error in the sway angle, estimated by $\int \varphi^2 \, \mathrm{d}t$, is therefore 1.8 times smaller than the DAE based error. Choosing different weighting matrices $\mathbf{Q}_{\mathrm{LQR}}$ and $\mathbf{R}_{\mathrm{LQR}}$ will yield different points on the pareto optimal curve.

Note that the necessary computation steps for the DAE based controller, such as linearisation, QR decomposition and control gain calculation is possible in real-time for any reference point. The controller is implemented with a time step size of 10 ms. For the shown experiments, the average computation time was 217 µs in each time step.

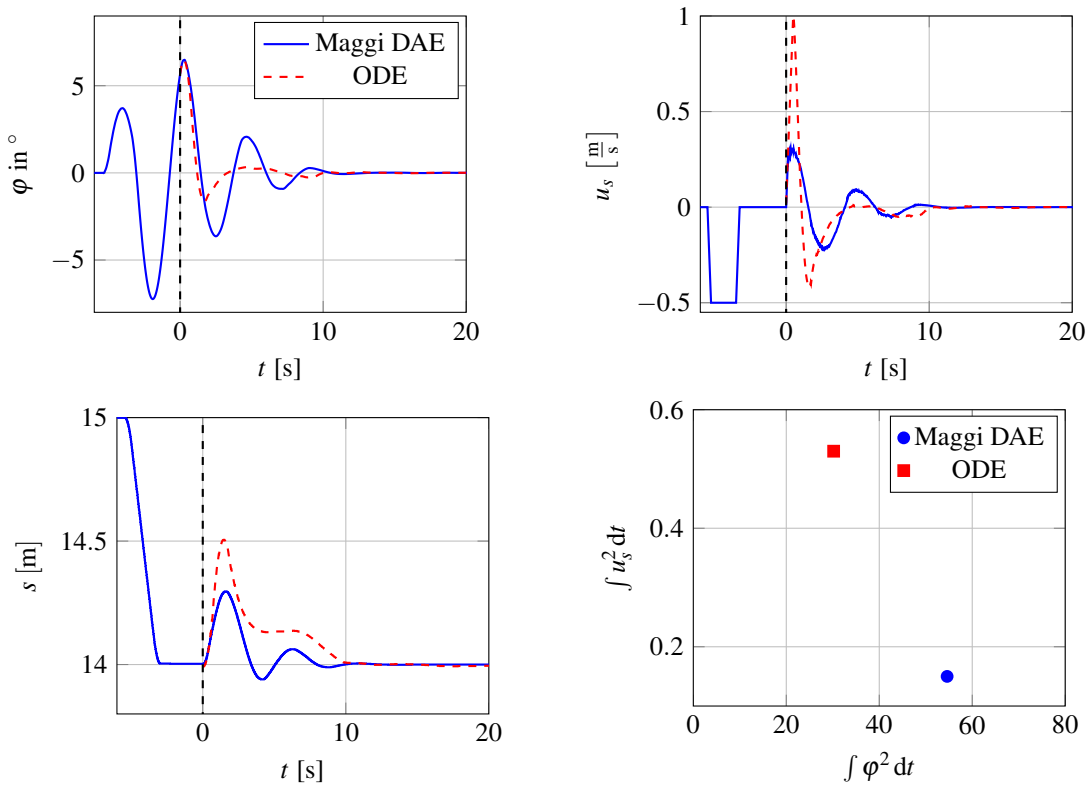

Figure 6 – Experimental results for platform sway reduction.

Trajectory Control

In order to emphasize the potential integration of the controller in a two-degree of freedom control structure, a combination of the state feedback controller with a feedforward controller is simulated for the model shown in Fig. 2. A straight trajectory from the initial position $\mathbf{z}_0 = \begin{bmatrix} 4\,\mathrm{m} & 4\,\mathrm{m} & 0° \end{bmatrix}^\top$ to the final position $\mathbf{z}_f = \begin{bmatrix} 6\,\mathrm{m} & 6\,\mathrm{m} & 0° \end{bmatrix}^\top$ is defined. An initial error $\Delta\dot{x}_{\mathrm{P}} = -0.5\,\frac{\mathrm{m}}{\mathrm{s}}$ is induced, indicating platform swaying motion φ. Note that Maggi's formulation with the QR decomposition is applied since both controllers showed similar results above. Thereby, the model is first linearised around the final position $\mathbf{z}_f$ and a linear time invariant version of the controller is applied. Then, the model is linearised

around the prescribed trajectory $\mathbf{z}_\mathrm{d}(t)$ yielding an linear time variant model and a linear time variant controller. In the two-degree of freedom control framework, the feedback controllers are added to the feedforward control $\mathbf{u}_\mathrm{d}$, such that the complete control arises as

$$\mathbf{u} = \mathbf{u}_d + \mathbf{K}(t)\left(\mathbf{x} - \mathbf{x}_\mathrm{d}(t)\right) . \tag{30}$$

Thereby, the feedback gain matrix $\mathbf{K}$ is either constant or time variant, depending on the LTI or LTV version respectively. The feedforward control is obtained from solving the servo-constraints DAE problem of Eqs. (1)-(2) and (10). As reference, the pure feedforward control $\mathbf{u}_\mathrm{d}$ is also applied to the disturbed system. The simulation results are shown in Fig. 7. The upper graph depicts the platform position in x,z coordinates. It shows that only using the feedforward control $\mathbf{u}_\mathrm{d}$ cannot dampen the sway motion and is not sufficient for control. Combining the state feedback controller with the feedforward control yields exact tracking. Thereby, a linear time invariant (LTI) version of the controller yields slightly larger tracking errors compared to the linear time variant (LTV) version obtained from linearising around the complete state trajectory. The lower left hand graph shows that the main control action is applied by the feedforward control $\mathbf{u}_\mathrm{d}$. The feedback part is only visible in the first few seconds until the disturbance $\Delta\dot{x}_\mathrm{P}$ is damped out. Note that it seems favourable for reaching the control objective to tilt the container while moving on the trajectory, see the upper right hand graph in Fig. 7. Note also that the linearisation around the desired trajectory $\mathbf{z}_\mathrm{d}(t)$ and solving of the respective differential Riccati equation for the LTV controller is very fast and therefore applicable for on-line implementation.

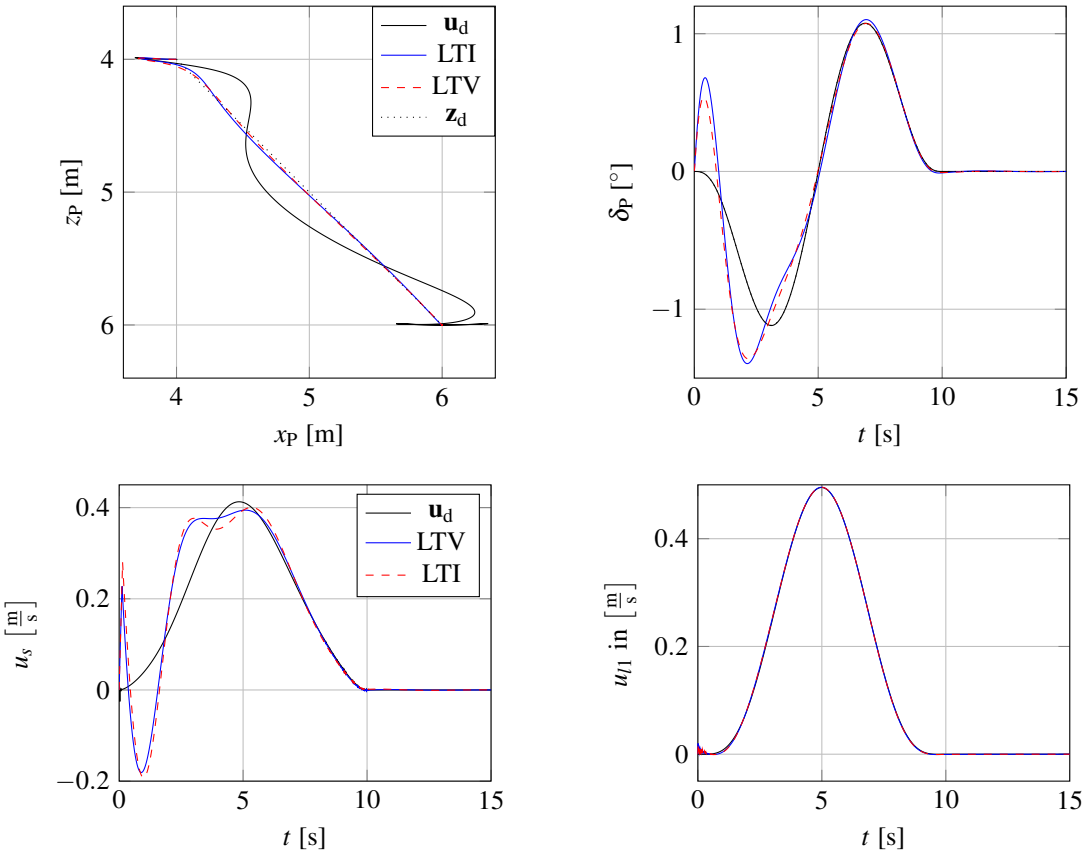

Figure 7 – Simulation of tracking a straight trajectory.

CONCLUSION

The DAE based controller applies Maggi's formulation and a QR decomposition of the projection matrices to transform the DAE model equations into ODEs. This transformation conserves the originally selected states and tuning of the controller weighting matrices is possible intuitively with the physical coordinates in mind. Moreover, implementation including linearisation, QR decomposition, and control gain calculation based on Riccati's equation is possible in real-time. In order to apply the proposed controller, design of a respective DAE based observer is necessary. The proposed UKF also applies the QR decomposition and reformulation of the DAE model into an ODE model. This strategy allows to include measurements of the algebraic coordinates and the algebraic constraints are fulfilled on velocity level. Two experiments verify the performance of the controller. First, platform orientation is controlled and compared to simulations. Second, damping of the platform sway motion is compared to a regular ODE based controller for a simplified model.

Moreover, the potential for combining the feedback controller with a feedforward controller is shown for tracking a straight trajectory in simulation.

REFERENCES

Bauchau, O. A., 2011, " Flexible Multibody Dynamics", Solid Mechanics and Its Applications

Bender, D., Laub, A., 1987, " The linear-quadratic optimal regulator for descriptor systems ", IEEE Transactions on Automatic Control, 32(8), pp. 672-688

Campbell, S.L. and Gear, C.W., 1995, " The index of general nonlinear DAEs", Numerische Mathematik 72(2), pp. 173-196

Findeisen, R. and Allgöwer, F, 2000 " Nonlinear Model Predictive Control for Index-one DAE Systems", Nonlinear Model Predictive Control. Progress in Systems and Control Theory, vol 26., pp. 145-161

Heiland, J., 2016, " A Differential-Algebraic Riccati Equation for Applications in Flow Control", SIAM Journal on Control and Optimization, 54(2), pp. 718-739

Julier, S.J. and Uhlmann, J.K., 2004; " Unscented filtering and nonlinear estimation", Proceedings of the IEEE, 92(3), pp. 401-422

Kim, S.S. and Vanderploeg, M.J., 1986; " QR decomposition for state space representation of constrained mechanical dynamic systems", Journal of mechanisms, transmissions, and automation in design, 108(2), pp. 183-188

Kreuzer, E., Pick, M.-A., Rapp, C. and Theis,J. , 2004; " Unscented kalman filter for real-time load swing estimation of container cranes using rope forces", Journal of Dynamic Systems, Measurement and Control, Transactions of the ASME, 136(4)

Mandela, R. K., Rengaswamy, R., Narasimhan, S. and Sridhar, L. N., 2010, " Recursive state estimation techniques for nonlinear differential algebraic systems", Chemical Engineering Science 65(16), pp. 4548-4556

Otto, S. and Seifried, R., 2018, " Real-time trajectory control of an overhead crane using servo-constraints", Multibody System Dynamics 42(1), pp. 1-17

Patwardhan, S. C., Narasimhan, S., Jagadeesan, P., Gopaluni, B. and Shah, S. L., 2012, " Nonlinear Bayesian state estimation: A review of recent developments", Control Engineering Practice , 20(10), pp. 933-953

Seifried, R., 2014, " Dynamics of Underactuated Multibody Systems Modeling, Control and Optimal Design", Springer International Publishing

Thrun, S., Burgard, W. and Fox, D., 2006, " Probabilistic robotics", MIT Press, Cambridge

ACKNOWLEDGEMENT

The project is supported by the German Research Foundation (Deutsche Forschungsgemeinschaft) via the grant SE 1685/6-1.

RESPONSIBILITY NOTICE

The authors are the only responsible for the printed material included in this paper.