

Real-time trajectory control of an overhead crane using servo-constraints

Svenja Otto¹ · Robert Seifried¹ 

Received: 27 September 2016 / Accepted: 17 February 2017 / Published online: 21 March 2017
© Springer Science+Business Media Dordrecht 2017

Abstract In this paper, the system dynamics of an overhead crane are inverted by servo-constraints. The inversion provides a feedforward control for trajectory tracking of the system output. The overhead crane is inherently underactuated and modeled as a two-dimensional mechanical system with nonlinear system dynamics. Actuators are modeled as first-order systems to simplify implementation and account for velocity-controlled drives. The control based on servo-constraints is shown to be an effective method of trajectory control for overhead cranes. It will be demonstrated that the formulation is solvable in real-time using linear implicit Euler integration. The feedforward solution is made robust by an augmentation with LQR as well as a sliding mode controller. Experiments are conducted on a laboratory crane of 13 m motion range.

Keywords Servo-constraints · Feedforward control · Overhead crane · Trajectory tracking · Underactuated systems

1 Introduction

Overhead cranes are typical underactuated multibody systems since they possess less control inputs than degrees of freedom. Specifically, they pose demanding control problems since the load swinging cannot be controlled directly. However, undesirable load swinging is easily excited during motion. Many control strategies have been investigated to maneuver the crane without inducing load swinging [1]. Due to the underactuated and nonlinear dynamics of the crane system, feedforward control for load trajectory tracking based on system inversion is not straightforward. The crane system analyzed in this paper is differentially flat [10]. Hence, an analytical feedforward control can be derived, which depends on

✉ R. Seifried
robert.seifried@tuhh.de

S. Otto
svenja.otto@tuhh.de

¹ Institute of Mechanics and Ocean Engineering, Hamburg University of Technology, Eißendorfer Straße 42, 21073 Hamburg, Germany

the desired output and a finite number of its derivatives. This property can also be used as a basis for feedback linearization, where for the crane system dynamic extension needs to be utilized and robustness issues need to be addressed [6]. However, the analytic derivation of the flatness based inversion or feedback linearization is not straightforward and might not be possible for more complex systems. An alternative and more general method of system inversion is the concept of servo-constraints. The servo-constraints append the system dynamics to form a set of differential algebraic equations (DAEs). A general framework considering servo-constraints is presented in [4]. The method has been applied to underactuated systems including cranes. Thereby it occurs that in crane problems often higher index DAEs appear [8]. An alternative servo-constraints formulation based on the system dynamics written in redundant coordinates is presented in [5]. Different methods for solving the resulting higher index DAEs have been presented recently. Most of them include an index reduction in the first step. Index reduction based on a coordinate projection and a coordinate transformation are introduced in [4] and [14], respectively. Index reduction based on minimal extension is proposed in [3]. For solving the reduced index problem, an implicit Euler scheme is proposed in [4] and [3].

The numerical solution of the servo-constraints formulation yields the feedforward control inputs for the system output to follow a prescribed trajectory. If the system model is accurate, the system output will move along the prescribed trajectory. In order to reject disturbances and account for modeling errors, additional feedback control is necessary. Therefore, a stabilized servo-constraints formulation is proposed in [4]. Alternatively, a two-degree of freedom controller design can be applied. For example, a simple feedback LQR controller could be added to the feedforward controller to stabilize the system around the prescribed trajectory [7]. Moreover, the servo-constraints solution could be used in combination with traditional sliding mode control [15], or other approaches such as [9]. The sliding mode control law requires information of the desired trajectory in state space. For underactuated systems, it is normally not straightforward to obtain the desired state trajectory based on the prescribed output trajectory. This normally makes the sliding mode controller difficult to apply. However, servo-constraints yield a simple and fast calculation of the desired trajectory in state space and enable a straightforward application of sliding mode control.

So far, the servo-constraints control approach has been investigated theoretically and verified by simulation. In this paper, it will be shown that the servo-constraints formulation can be solved by a linear implicit Euler method in real-time. Thus, the scheme can be used as an online feedforward controller which could allow online adaption of the desired load trajectory. The effective combination of the servo-constraints controller with an LQR as well as a sliding mode controller is also shown. All results are verified on an experimental test bench of an overhead crane with 13 m motion range.

The remainder of the paper is organized as follows. In Sect. 2, the crane model is derived in generalized and redundant coordinates. The servo-constraints problem is formulated in Sect. 3 for both coordinate formulations. A solution code for the formulated problem is proposed in Sect. 4. An LQR and a sliding mode controller that can be combined with the servo-constraints inversion are presented in Sect. 5. Section 6 presents experimental results including analysis of the real-time capability of the various numerical formulations and the accuracy of the different control strategies. Section 7 summarizes the paper.

2 Modeling of the crane test bench

The crane test bench is an overhead gantry crane with variable rope length, shown in Fig. 1(a). Since the trolley moves along a linear axis, planar motion is assumed. Each of

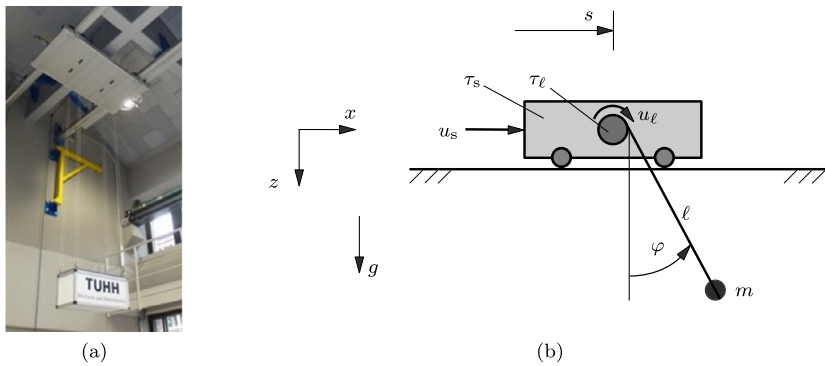


Fig. 1 Laboratory crane setup and its model. (a) Test bench. (b) Two-dimensional crane model

the four ropes connecting the load to the trolley is assumed to be massless and rigid. The rope lengths are adjustable by synchronized winches on the trolley, which prevents the load from rotating around itself. Thus, the model can be reduced to a point mass which is attached to the trolley by a single rope, shown in Fig. 1(b). The model has $n = 3$ degrees of freedom, namely the trolley position s , the rope length ℓ and the swing angle φ . They are collected in the vector of generalized coordinates \mathbf{y} and velocities \mathbf{v}

$$\mathbf{y} = \begin{bmatrix} \mathbf{y}_a \\ \mathbf{y}_u \end{bmatrix} = \begin{bmatrix} s \\ \ell \\ \varphi \end{bmatrix}, \quad \mathbf{v} = \dot{\mathbf{y}} = \begin{bmatrix} \dot{\mathbf{y}}_a \\ \dot{\mathbf{y}}_u \end{bmatrix} = \begin{bmatrix} \dot{s} \\ \dot{\ell} \\ \dot{\varphi} \end{bmatrix}, \tag{1}$$

where the dashed lines denote the separation into actuated and unactuated coordinates \mathbf{y}_a , \mathbf{v}_a and \mathbf{y}_u , \mathbf{v}_u , respectively. The $m = 2$ system inputs are realized by trolley and winch actuators. Due to $m < n$, the system is underactuated. In theory, the control inputs are the actuator force on the trolley and actuator torque on the winch. However, force and torque controlled actuators are difficult to realize and often pose robustness issues due to drive train friction. Therefore, the test bench actuators are velocity-controlled. Hence, the system control inputs u_s and u_ℓ are modeled as reference velocities of trolley and rope length, respectively. The rope velocity $\dot{\ell}$ and trolley velocity \dot{s} are then rheonomic constraints on the system. This accelerates implementation and yields a more robust control design. The inputs are collected in the vector

$$\mathbf{u} = \begin{bmatrix} u_s \\ u_\ell \end{bmatrix}. \tag{2}$$

The actuator dynamics are modeled as first-order dynamics

$$\tau_s \ddot{s} + \dot{s} = u_s, \tag{3}$$

$$\tau_\ell \ddot{\ell} + \dot{\ell} = u_\ell, \tag{4}$$

with the time constants τ_s and τ_ℓ of trolley and winch actuator, respectively. The time constants can be identified experimentally on the test bench.

The equations of motion of a multibody system arise in the form

$$\mathbf{M}(\mathbf{y}, t)\dot{\mathbf{v}} + \mathbf{k}(\mathbf{y}, \mathbf{v}, t) = \mathbf{q}(\mathbf{y}, \mathbf{v}, t) + \mathbf{B}\mathbf{u}, \tag{5}$$

with the generalized mass matrix \mathbf{M} , the vector of Coriolis, centrifugal and gyroscopic forces \mathbf{k} , the vector of applied forces \mathbf{q} and the distribution matrix of the system inputs \mathbf{B} . For underactuated multibody systems, Eq. (5) can be split into an actuated and unactuated parts

$$\begin{bmatrix} \mathbf{M}_{aa} & \mathbf{M}_{au} \\ \mathbf{M}_{ua} & \mathbf{M}_{uu} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{v}}_a \\ \dot{\mathbf{v}}_u \end{bmatrix} + \begin{bmatrix} \mathbf{k}_a \\ \mathbf{k}_u \end{bmatrix} = \begin{bmatrix} \mathbf{q}_a \\ \mathbf{q}_u \end{bmatrix} + \begin{bmatrix} \mathbf{B}_a \\ \mathbf{0} \end{bmatrix} \mathbf{u}. \tag{6}$$

For the considered crane, the equation of motion (5) is derived with the Newton–Euler formalism for the swing angle φ , under rheonomic constraints of rope and trolley velocities. Adding the actuator models of Eq. (3) and Eq. (4) yields the equations

$$\begin{bmatrix} \tau_s & 0 & 0 \\ 0 & \tau_\ell & 0 \\ \cos \varphi & 0 & \ell \end{bmatrix} \begin{bmatrix} \ddot{s} \\ \ddot{\ell} \\ \ddot{\varphi} \end{bmatrix} + \begin{bmatrix} \dot{s} \\ \dot{\ell} \\ 2\dot{\varphi}\dot{\ell} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -g \sin \varphi \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \mathbf{u}, \tag{7}$$

where the dashed lines denote the separation into actuated and unactuated parts. Due to the inclusion of the actuator dynamics as first-order systems, the matrix \mathbf{M} is not symmetric anymore but is still non-singular. Note that the equations of motion only depend on the time constants τ_s and τ_ℓ and the gravitational constant g . They depend on neither trolley mass nor load mass. The output z of the crane model is defined as the load position

$$\mathbf{z} = \mathbf{h}(\mathbf{y}) = \begin{bmatrix} s + \ell \sin(\varphi) \\ \ell \cos(\varphi) \end{bmatrix}. \tag{8}$$

Alternatively to using minimal coordinates, a set of redundant coordinates \mathbf{y}_r in combination with constraint equations can be used to describe the crane motion. This yields a system of differential-algebraic equations

$$\mathbf{M}_r(\mathbf{y}_r, t) \dot{\mathbf{v}}_r + \mathbf{k}_r(\mathbf{y}_r, \mathbf{v}_r, t) = \mathbf{q}_r(\mathbf{y}_r, \mathbf{v}_r, t) + \mathbf{Q}^T \boldsymbol{\lambda} + \mathbf{B}_r \mathbf{u}, \tag{9}$$

$$\mathbf{c}(\mathbf{y}_r) = \mathbf{0}, \tag{10}$$

where the vector $\boldsymbol{\lambda}$ contains the Lagrangian multipliers which arise to fulfill the geometric constraints $\mathbf{c}(\mathbf{y}_r)$ [13]. The Jacobian of the constraints is

$$\mathbf{Q} = \frac{\partial \mathbf{c}(\mathbf{y}_r)}{\partial \mathbf{y}_r}. \tag{11}$$

For the presented system, it is convenient to describe the load dynamics in redundant coordinates. Then, the complete vector of redundant coordinates is

$$\mathbf{y}_r = \begin{bmatrix} \mathbf{y}_{r,a} \\ \mathbf{y}_{r,u} \end{bmatrix} = \begin{bmatrix} s \\ \ell \\ x_2 \\ z_2 \end{bmatrix}, \quad \dot{\mathbf{y}}_r = \mathbf{v}_r = \begin{bmatrix} \dot{\mathbf{y}}_{r,a} \\ \dot{\mathbf{y}}_{r,u} \end{bmatrix} = \begin{bmatrix} \dot{s} \\ \dot{\ell} \\ \dot{x}_2 \\ \dot{z}_2 \end{bmatrix}, \tag{12}$$

where x_2 and z_2 describe the load position in the coordinate system depicted in Fig. 1(b). Once again replacing the trolley dynamics by the actuator dynamics of Eq. (3) and Eq. (4)

yields the crane dynamics in DAE form

$$\begin{bmatrix} \tau_s & 0 & 0 & 0 \\ 0 & \tau_\ell & 0 & 0 \\ \hline 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \ddot{s} \\ \ddot{\ell} \\ \ddot{x}_2 \\ \ddot{z}_2 \end{bmatrix} + \begin{bmatrix} \dot{s} \\ \dot{\ell} \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ g \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \hline 2(x_2 - s) \\ 2z_2 \end{bmatrix} \lambda + \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ \hline 0 & 0 \\ 0 & 0 \end{bmatrix} \mathbf{u}, \tag{13}$$

$$z_2^2 + (x_2 - s)^2 - \ell^2 = 0, \tag{14}$$

where the only Lagrangian multiplier represents the rope force F_S to fulfill the rigid rope constraint (14). The system output in redundant coordinates simply is

$$\mathbf{z} = \mathbf{h}_r(\mathbf{y}_r) = \mathbf{y}_{r,u} = \begin{bmatrix} x_2 \\ z_2 \end{bmatrix}. \tag{15}$$

The control objective is to move the load along a trajectory prescribed in time and space. An inverse model is derived in the following section to accomplish trajectory tracking.

3 Servo-constraints problem formulation

Servo-constraints impose additional constraints on the system formulations of either Eq. (7) or Eqs. (13) and (14) by enforcing the desired output trajectory $\mathbf{z}_d(t)$. The servo-constraints therefore are

$$\mathbf{0} = \mathbf{h}(\mathbf{y}) - \mathbf{z}_d(t), \tag{16}$$

where both coordinate definitions can be used. The servo-constraints Eq. (16) extend the equations of motion to form a system of DAEs

$$\dot{\mathbf{y}} = \mathbf{v}, \tag{17}$$

$$\mathbf{M}(\mathbf{y}, t)\dot{\mathbf{v}} + \mathbf{k}(\mathbf{y}, \mathbf{v}, t) = \mathbf{q}(\mathbf{y}, \mathbf{v}, t) + \mathbf{Q}^T(\mathbf{y})\lambda + \mathbf{B}\mathbf{u}, \tag{18}$$

$$\mathbf{0} = \mathbf{h}(\mathbf{y}) - \mathbf{z}_d(t), \tag{19}$$

$$\mathbf{0} = \mathbf{c}(\mathbf{y}), \tag{20}$$

for both formulations in generalized coordinates (1) and redundant coordinates (12). The DAE setup in generalized coordinates will be abbreviated GC, while the formulation using redundant coordinates will be abbreviated RC in the following. In case of GC, the geometric constraints $\mathbf{c}(\mathbf{y})$ vanish. Note that the dependencies of the matrices on the state vectors are omitted due to readability reasons in the following.

3.1 Formulation in generalized coordinates

For the crane system in generalized coordinates, the DAE system of Eqs. (17)–(20) is of index 5 and will be referred to as GC5 formulation. It is analyzed in detail in [4], where the formulation includes torque and force inputs in contrast to the reference velocity inputs used here. The inclusion of actuator dynamics does not change the index of the DAE. A projection approach is proposed in [4] to reduce the formulation to an index 3 formula-

tion. Equations (17)–(19) are projected into a constrained and unconstrained subspace. The constrained subspace is spanned by the Jacobian of the output

$$C = \frac{\partial \mathbf{h}}{\partial \mathbf{y}}, \tag{21}$$

while the unconstrained subspace is constructed as a complementary subspace. The unconstrained subspace is spanned by the matrix D , which is constructed by demanding

$$CD = \mathbf{0}. \tag{22}$$

Following [4], a premultiplication of the DAE system in Eqs. (17)–(19) by the matrix $[D^T \ C M^{-1}]^T$ yields a new set of index 3 DAEs

$$\dot{\mathbf{y}} = \mathbf{v}, \tag{23}$$

$$D^T M \dot{\mathbf{v}} = D^T (\mathbf{q} - \mathbf{k}) + D^T \mathbf{B} \mathbf{u}, \tag{24}$$

$$\mathbf{0} = C M^{-1} (\mathbf{q} - \mathbf{k}) + C M^{-1} \mathbf{B} \mathbf{u} - \ddot{\mathbf{z}}_d + \dot{C} \mathbf{v}, \tag{25}$$

$$\mathbf{0} = \mathbf{h} - \mathbf{z}_d. \tag{26}$$

This formulation will be referred to as GC3 formulation. Note that here, Eq. (23) is of dimension $n = 3$, Eq. (24) of dimension $n - m = 1$, and Eqs. (25) and (26) are of dimension $m = 2$. The total number of equations, $2n + m = 8$, matches the number of unknowns, which are the desired state states \mathbf{y}_d , its derivatives \mathbf{v}_d , and the corresponding inputs \mathbf{u}_d . The desired input trajectory \mathbf{u}_d poses the feedforward control input.

3.2 Formulation in redundant coordinates

In redundant coordinate formulation, the DAE system of Eqs. (17)–(20) is also of index 5 and will be referred to as RC5 formulation. It is even more straightforward to set up the servo-constraints

$$\mathbf{0} = \mathbf{y}_{r,u} - \mathbf{z}_d, \tag{27}$$

because the load position is directly part of the coordinate vector. It was proposed in [5] to substitute Eq. (27) into the original crane dynamics of Eqs. (13)–(14), yielding

$$\dot{\mathbf{y}}_{r,a} = \mathbf{v}_{r,a}, \tag{28}$$

$$\begin{bmatrix} \tau_s & 0 & 0 & 0 \\ 0 & \tau_\ell & 0 & 0 \\ \hline 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \ddot{s} \\ \ddot{\ell} \\ \ddot{x}_{2d} \\ \ddot{z}_{2d} \end{bmatrix} + \begin{bmatrix} \dot{s} \\ \dot{\ell} \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ g \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \hline 2(x_{2d} - s) \\ 2z_{2d} \end{bmatrix} \boldsymbol{\lambda} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ \hline 0 & 0 \\ 0 & 0 \end{bmatrix} \mathbf{u}, \tag{29}$$

$$z_{2d}^2 + (x_{2d} - s)^2 - \ell^2 = 0. \tag{30}$$

The last two equations of Eq. (29) are now algebraic instead of differential equations, and the original servo-constraints are removed from the set of equations. This reduces the number of equations and unknowns by $m = 2$ equations. Note that the substitution of the servo-constraints already yields an index 3 DAE which will be referred as RC3. Another projection is not necessary.

The approach of using redundant coordinates has some advantages since the derivation of the equations of motion is easier and a projection to obtain an index 3 DAE is not necessary in this case. Moreover, the rope force F_S is contained in Eq. (29) in terms of λ and is therefore one of the unknowns of the system. This is convenient for checking feasible trajectories by demanding $F_S(t) > 0 \forall t$. On the other hand, the servo-constraints solution does not contain the desired generalized coordinates y_d anymore. If necessary for feedback control, it has to be generated from the desired output z_d and desired redundant coordinates $y_{r,d}$ during post-processing.

4 Solution code

In Sect. 3, four servo-constraints formulations were introduced, namely index 5 formulations in generalized (GC5) and redundant coordinates (RC5) and index 3 formulations in generalized (GC3) and redundant coordinates (RC3). All formulations can be solved by numerical schemes. For notational simplicity, the DAE system is collected in the vector

$$F(v, y, u, \dot{v}, \dot{y}, \lambda) = 0 \tag{31}$$

for all formulations. Different numerical schemes can be applied to solve Eq. (31) [11]. An implicit Euler scheme is applied here as also proposed in [4] for such servo-constraints problem. All derivatives at time t_{k+1} are approximated by finite differences

$$\dot{y}_{k+1} \approx \frac{y_{k+1} - y_k}{\Delta t}, \tag{32}$$

where the indices $k + 1$ and k denote the corresponding values at time t_{k+1} and time t_k , respectively. In total, a set of nonlinear equations arises as follows:

$$F\left(v_{k+1}, y_{k+1}, u_{k+1}, \frac{v_{k+1} - v_k}{\Delta t}, \frac{y_{k+1} - y_k}{\Delta t}, \lambda_{k+1}\right) = 0, \tag{33}$$

which is solved for the solution vector

$$x_{k+1} = \begin{bmatrix} v_{k+1} \\ y_{k+1} \\ u_{k+1} \\ \lambda_{k+1} \end{bmatrix} \tag{34}$$

at each time step t_{k+1} . Note that for GC, the vector λ_{k+1} vanishes. Newton’s method is applied to solve the nonlinear equations in Eq. (33). The solution of the previous time step x_k is chosen as an initial guess x_{k+1}^0 for the next iteration. Theoretically, convergence of Newton’s method within the available control loop step time cannot be assured. Therefore, only one Newton iteration is proposed to solve Eq. (33). This is equivalent to the linear implicit Euler scheme [11]. In Sect. 6, the accuracy of the linear implicit Euler scheme is compared to the implicit Euler scheme. For the implicit Euler scheme, the convergence criterion of Newton’s method is chosen as the absolute criterion

$$\|F\|_{\max} < 1.0 \cdot 10^{-9}. \tag{35}$$

However, to ensure fast computation, Newton's method is stopped after a maximum of i_{\max} iterations. Computational speed is further increased by symbolically precalculating the Jacobian

$$\mathbf{J}_F = \frac{\partial \mathbf{F}}{\partial \mathbf{x}_{k+1}} = \begin{bmatrix} \frac{\partial F}{\partial v_{k+1}} & \frac{\partial F}{\partial y_{k+1}} & \frac{\partial F}{\partial u_{k+1}} & \frac{\partial F}{\partial f_{r,k+1}^T} \end{bmatrix}. \quad (36)$$

Real-time applicability of the proposed scheme will be investigated in detail in Sect. 6.

5 Stabilizing control

If the system model represented the overhead crane perfectly and no disturbances occurred, the feedforward input \mathbf{u}_d would be sufficient for trajectory tracking. However, due to model errors and disturbances, feedback control is necessary to stabilize the system around the prescribed trajectory. The feedback channel makes use of the desired state trajectories \mathbf{y}_d and \mathbf{v}_d which result from the servo-constraints solution. A linear quadratic regulator (LQR) and a sliding mode controller are designed for stabilizing.

In order to apply the LQR optimization problem, the cost function J is defined as

$$J = \frac{1}{2} \int_{t_0}^{\infty} \mathbf{x}(t)^T \mathbf{Q} \mathbf{x}(t) + \mathbf{u}(t)^T \mathbf{R} \mathbf{u}(t) dt, \quad (37)$$

where $\mathbf{Q} \in \mathbb{R}^{6 \times 6}$ and $\mathbf{R} \in \mathbb{R}^{2 \times 2}$ are weighting matrices used for tuning [7]. The system dynamics are linearized around the final state of the trajectory to apply the LQR. The obtained gains are then used for the entire trajectory. Solving the LQR problem yields a linear time invariant gain matrix $\mathbf{K} \in \mathbb{R}^{m \times 2n}$. It is used in the feedback law

$$\mathbf{u}_{\text{LQR}} = \mathbf{K} \begin{bmatrix} \tilde{\mathbf{y}} \\ \tilde{\mathbf{v}} \end{bmatrix}, \quad (38)$$

where the tilde denotes the error between desired and measured trajectory of the generalized coordinates

$$\tilde{\mathbf{y}} = \begin{bmatrix} \tilde{\mathbf{y}}_a \\ \tilde{\mathbf{y}}_u \end{bmatrix} = \mathbf{y} - \mathbf{y}_d, \quad \tilde{\mathbf{v}} = \mathbf{v} - \mathbf{v}_d. \quad (39)$$

The feedback control \mathbf{u}_{LQR} accounts for small variations around the prescribed trajectory. The main control action for following the trajectory is provided by the feedforward control \mathbf{u}_d . This results in less noise amplification in the feedback loop. Alternatively to the constant gain matrix, one might use a linearization around the prescribed trajectory \mathbf{y}_d . This yields a linear time variant system for each trajectory. Solving the corresponding Riccati differential equation yields a time variant gain matrix $\mathbf{K}(t)$. A comparison with the simple time-invariant approach shows no advantage for the presented system [12].

Additionally, a sliding mode controller is compared to the LQR controller. The sliding mode controller is set up for the underactuated multibody system as proposed in [2]. A sliding surface s of dimension $n = 2$ is defined as

$$s = \Theta_a \tilde{\mathbf{v}}_a + \Lambda_a \tilde{\mathbf{y}}_a + \Theta_u \tilde{\mathbf{v}}_u + \Lambda_u \tilde{\mathbf{y}}_u \quad (40)$$

$$= \Theta_a \mathbf{v}_a + \Theta_u \mathbf{v}_u - \underbrace{\Theta_a \mathbf{v}_{d,a} - \Theta_u \mathbf{v}_{d,u}}_{s_p} + \Lambda_a \tilde{\mathbf{y}}_a + \Lambda_u \tilde{\mathbf{y}}_u. \quad (41)$$

The surface parameters $\Theta_a, \Lambda_a \in \mathbb{R}^{2 \times 2}$, and $\Theta_u, \Lambda_u \in \mathbb{R}^{2 \times 1}$ are chosen to yield stable sliding mode surfaces [2]. Stable sliding mode surfaces can, for example, also be obtained by singular LQ surface design; see [9]. The control law for the sliding mode input u_{SMC} is obtained from $\dot{s} = \mathbf{0}$, so that

$$u_{SMC} = -M_s^{-1}(f_s + \dot{s}_p + \kappa \text{sat}(s\Phi^{-1})), \tag{42}$$

where the matrices M_s and f_s are defined as

$$M_s = \Theta_a M'_{aa}{}^{-1} - \Theta_u M'_{uu}{}^{-1} M_{ua} M'_{aa}{}^{-1}, \tag{43}$$

$$f_s = \Theta_a M'_{aa}{}^{-1} f'_a + \Theta_u M'_{uu}{}^{-1} f'_u, \tag{44}$$

and

$$M'_{aa} = M_{aa} - M_{au} M_{uu}^T M_{ua}, \tag{45}$$

$$f'_a = (q_a - k_a) - M_{au} M_{uu}^T (q_u - k_u), \tag{46}$$

$$M'_{uu} = M_{uu} - M_{ua} M_{aa}^T M_{au}, \tag{47}$$

$$f'_u = (q_u - k_u) - M_{ua} M_{aa}^T (q_a - k_a). \tag{48}$$

Hereby the partition of the equations of motion (6) is used. For the detailed derivation, see [2]. The gains κ in Eq. (42) are chosen to ensure that the sliding surface is reached in finite time [2, 15]. The width Φ of the saturation function is tuned to minimize the error while avoiding chattering. The desired states y_d, \dot{y}_d , and \ddot{y}_d are necessary to calculate Eq. (42). They are obtained from solving the servo-constraints formulation. The second derivatives of the state trajectory \ddot{y}_d are obtained by finite differences

$$\ddot{y}_{d,k+1} \approx \frac{v_{d,k+1} - v_{d,k}}{\Delta t}. \tag{49}$$

Both the LQR feedback control u_{LQR} in combination with the feedforward law u_d and the sliding mode control u_{SMC} are tested in experiments.

6 Experimental results

The crane test bench is set up at the Institute of Mechanics and Ocean Engineering at Hamburg University of Technology and is depicted in Fig. 1(a). The nominal physical parameters are listed in Table 1. Trolley and winches are actuated by synchronous servo-motors. The motion range lies within 13 m and 9 m for trolley and ropes, respectively. Additional kinematic constraints are shown in Table 2.

A LABVIEW program running on a real-time computer controls the test bench. The real-time computer runs a 32 bit version of the NATIONAL INSTRUMENTS real-time operating system PHAR LAB 13.1. The LABVIEW control loop is running with a time step size of $\Delta t = 10$ ms. Absolute angle encoders measure the current trolley position s and the rope length ℓ . The swing angle is estimated by an unscented Kalman filter based on measurements of rope length and rope force. The estimated angle $\hat{\varphi}$, the measured trolley position \hat{s} and

Table 1 Physical parameters of the container crane test bench

Parameter	Nominal value
load dimension	$(0.35 \times 0.37 \times 0.86)$ m
load mass	$m = 12.9$ kg
trolley mass	$m_T = 270$ kg
time constant trolley	$\tau_s = 0.03$ s
time constant winch	$\tau_\ell = 0.02$ s

Table 2 Kinematic parameters of the container crane test bench

Parameter	Admissible range
swing angle	$\varphi \in [-0.2, 0.2]$ rad
angular velocity	$\dot{\varphi} \in [-0.25, 0.25]$ $\frac{\text{rad}}{\text{s}}$
rope length	$\ell \in [3, 12]$ m
rope velocity	$\dot{\ell} \in [-2, 2]$ $\frac{\text{m}}{\text{s}}$
trolley position	$s \in [0, 13]$ m
trolley velocity	$\dot{s} \in [-3, 3]$ $\frac{\text{m}}{\text{s}}$

measured rope length $\hat{\ell}$ determine the estimated load position \hat{z} by

$$\hat{z} = \begin{bmatrix} \hat{s} + \hat{\ell} \sin(\hat{\varphi}) \\ \hat{\ell} \cos(\hat{\varphi}) \end{bmatrix}. \tag{50}$$

For control performance evaluation, the estimated load trajectory $\hat{z}(t)$ is compared to the desired trajectory $z_d(t)$. In the experiments, the desired trajectory is a straight path or semicircle defined by an initial and final load position $p_0 = [x_{20} \ y_{20} \ z_{20}]^T$ and $p_f = [x_{2f} \ y_{2f} \ z_{2f}]^T$, respectively, defined in the coordinate system depicted in Fig. 1(b). Every point $p(s)$ on the straight path is parameterized by the scalar σ so that

$$p(\sigma) = p_0 + \frac{\sigma}{\sigma_t}(p_f - p_0), \quad \sigma \in [0, \sigma_t], \tag{51}$$

where the parameter σ_t is the total arc length $\sigma_t = \|p_f - p_0\|_2$. The temporal sequence is defined by a timing law $\sigma = \sigma(t)$. The parameter $\sigma(t)$ must be four times continuously differentiable and is given by

$$\sigma(t) = \left(126 \left(\frac{t}{t_f}\right)^5 - 420 \left(\frac{t}{t_f}\right)^6 + 540 \left(\frac{t}{t_f}\right)^7 - 315 \left(\frac{t}{t_f}\right)^8 + 70 \left(\frac{t}{t_f}\right)^9 \right) \sigma_t, \tag{52}$$

where the initial time is $t_0 = 0$ s and the transition time is t_f ; see, e.g., [4].

6.1 Analysis of servo-constraints solution

Firstly, the solution code proposed in Sect. 4 is analyzed with respect to its accuracy. For this purpose, the servo-constraints solution is compared to the analytical feedforward solution u_{flat} based on differential flatness [10]. The trajectory error $e(t)$ is defined as the largest difference of the feedforward control inputs at each time instant

$$e(t) = \max_i \|u_{d,i}(t) - u_{\text{flat},i}(t)\|. \tag{53}$$

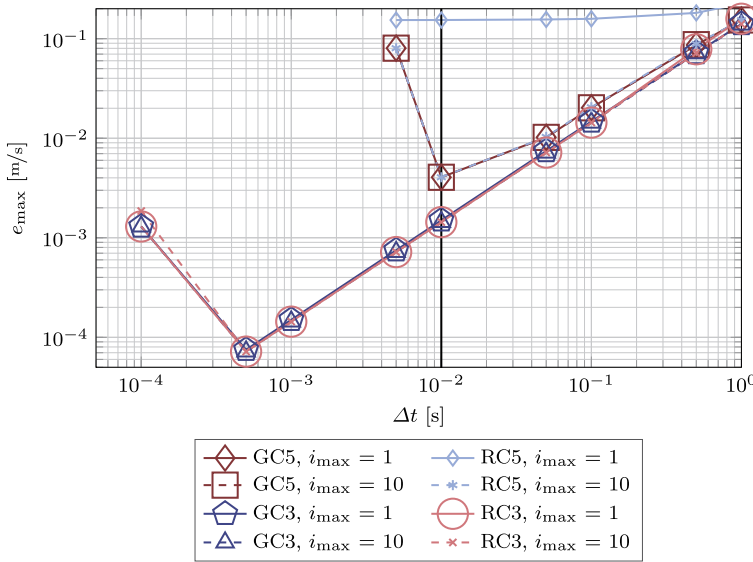


Fig. 2 Maximum error e_{max} between numerical and analytical solution for different time step sizes Δt

The maximum error e_{max} is the maximum of the error $e(t)$ over time

$$e_{max} = \max_t \left(\max_i \|u_{d,i}(t) - u_{flat,i}(t)\| \right) = \max_t (e(t)). \tag{54}$$

Both formulations in generalized and redundant coordinates are solved in index 3 and index 5 configuration. All four formulations are solved with the linear implicit Euler ($i_{max} = 1$) and the implicit Euler with at most $i_{max} = 10$ Newton iterations. The simulations are performed for a linear trajectory with initial position $p_0 = [15 \ 0 \ 4]^T$ m, final position $p_f = [11 \ 0 \ 7]^T$ m and transition time $t_f = 10$ s.

First-order convergence of the implicit Euler method is demonstrated in Fig. 2, where the maximum error e_{max} is plotted over the step size Δt . For all formulations except RC5, the maximum error e_{max} of the linear implicit Euler scheme is comparable to the error of the implicit Euler scheme with possibly more than one iteration. Therefore, the linear implicit Euler scheme is applicable on the test bench. This ensures real-time applicability. Moreover, all index 3 formulations and all index 5 formulations, except for RC5 with $i_{max} = 1$, yield comparable errors e_{max} , respectively. The index 5 formulations run into numerical rounding errors for time step smaller than $\Delta t = 10$ ms. Both formulations RC3 and GC3 are stable for time steps larger than $\Delta t = 0.5$ ms for the simulated scenario. The results show that a time step $\Delta t = 10$ ms, which is available on the test bench, is sufficient for the accuracy of the numerical solution in both index 3 formulations. The maximum error $e_{max} \approx 1 \cdot 10^{-3}$ m/s is small compared to inaccuracies, e.g., due to actuator friction. This justifies the usage of the Euler scheme. Note that since the system is differentially flat, numerical effects such as numerical damping do not significantly influence the solution. Therefore, both index 3 formulations GC3 and RC3 are used for experiments.

Note that the maximum error e_{max} does not grow over integration time. There is no drift-off effect of the algebraic variables, since the servo-constraints are fulfilled exactly at position level in each integration step. This is demonstrated in Fig. 3. This is an important property concerning long-term applicability.

Fig. 3 Error $e(t)$ over time for both index 3 formulations with time step $\Delta t = 10$ ms

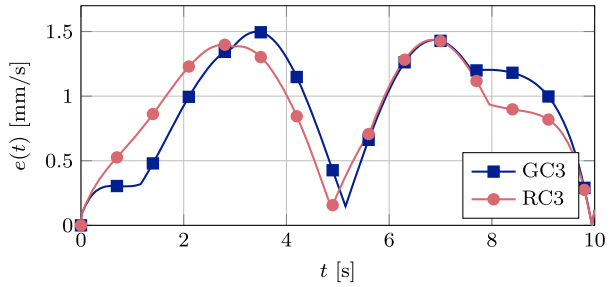


Table 3 Control parameters of the feedback and sliding mode control, where I_i denotes the identity matrix of size i

Parameter	R	Q	Θ_a	Λ_a	Θ_u	Λ_u	κ	Φ
Value	$100I_2$	$\begin{bmatrix} 1000I_3 & 0 \\ 0 & 1000I_3 \end{bmatrix}$	I_2	$0.5I_2$	$\begin{bmatrix} 1.5 \\ 0 \end{bmatrix}$	$\begin{bmatrix} -5.0 \\ 0 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 1 \end{bmatrix}$	I_2

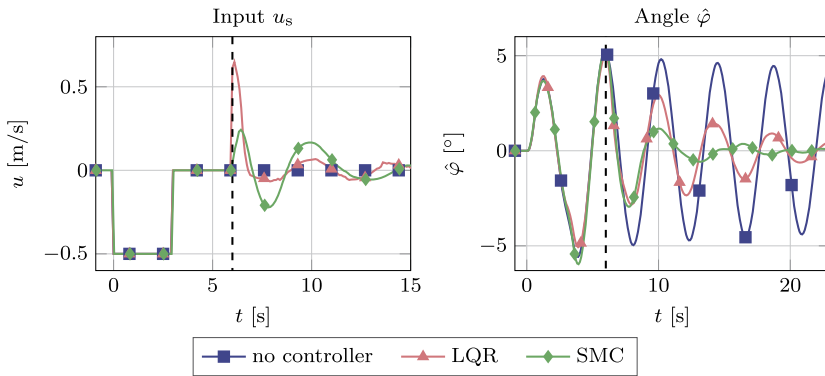


Fig. 4 Experimental results for the load swing reduction experiment

6.2 Experiments

In experiments, the pure servo-constraints feedforward controller (SC), the feedforward controller stabilized by LQR (SC+LQR), and the sliding mode controller (SMC) proposed in Sect. 5 are compared. Note that for the servo-constraints feedforward controller, both formulations in GC3 and RC3 are implemented and compared. It is also noted that the SMC uses the desired state trajectories y_d, v_d from the servo-constraints inverse model which is also solved for the SMC online. The tuning parameters of LQR and SMC are shown in Table 3. The stability properties of both LQR and SMC are demonstrated independently from the feedforward loop in the first experiment. For this purpose, the trolley is disturbed with a constant step velocity $u_s = -0.5$ m/s for 3 s. This induces load swinging. After a waiting time of another 3 s, the feedback controllers are turned on to dampen the induced load swinging of $\varphi_0 \approx 5.2^\circ$ and bring the system to rest. The input u and estimated angle $\hat{\varphi}$ are shown in Fig. 4. The corresponding load swinging of the free system is also shown. Both feedback controllers are able to dampen the load swinging efficiently compared to the

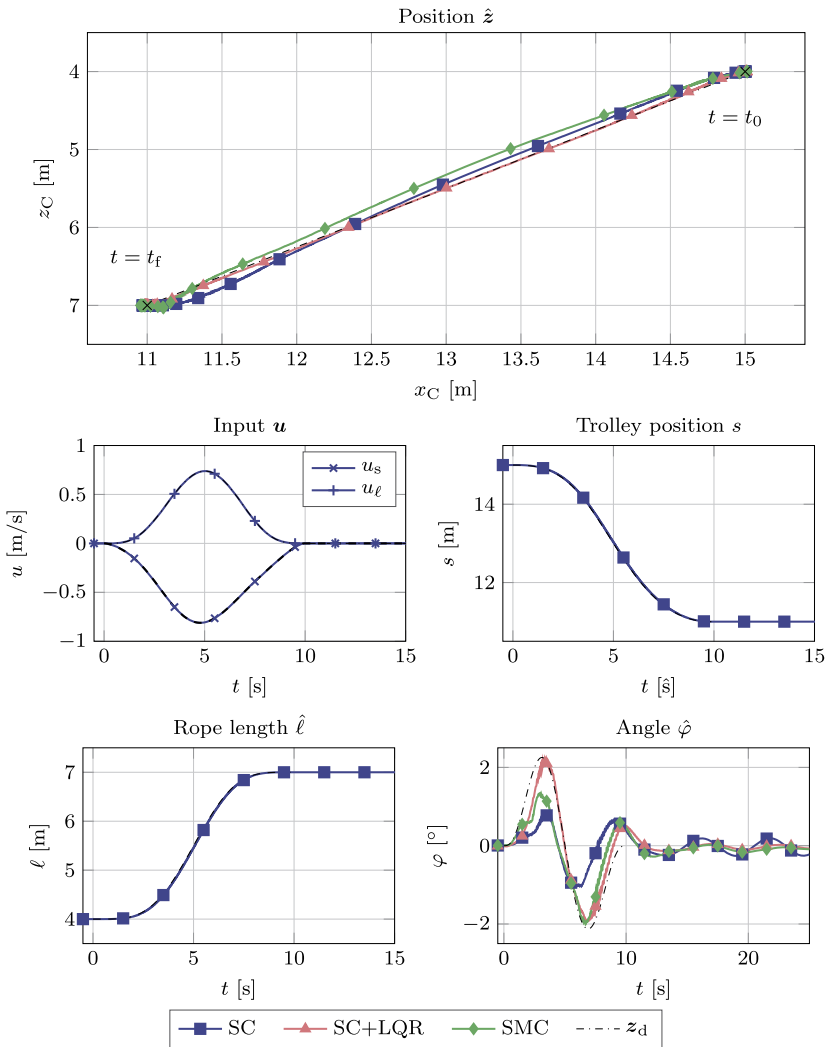


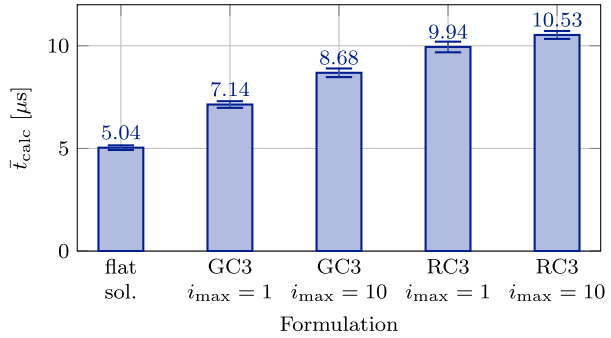
Fig. 5 Experimental results for the straight trajectory 1

free system. The sliding mode controller shows superior performance compared to the LQR controller. It reduces load swinging much faster.

After verification of the stabilizing properties of the feedback controller, trajectory tracking is analyzed. For the straight trajectory defined in Sect. 6.1, called trajectory 1, the estimated load position \hat{z} and the input and state trajectories are shown in Fig. 5. All proposed controllers are able to follow the prescribed trajectory. Note that only the angle $\hat{\varphi}$ is shown for all three experiments because the other trajectories are not distinguishable in the plots.

In terms of numerical accuracy, it is shown in Fig. 2 that both formulations GC3 and RC3 yield similar numerical errors. This results in similar performance in terms of tracking accuracy. Regarding real-time capability, the calculation time $t_{calc,k}$ which is necessary to solve the servo-constraints formulation in each time step k is measured on the real-time

Fig. 6 Mean and twice the standard deviation of 10 measurements each of the mean calculation time \bar{t}_{calc} for each time step for trajectory 1 with transition time $t_f = 10$ s



target with a resolution of 1 μs for both Gc3 and RC3. The mean calculation time for each trajectory is calculated by

$$\bar{t}_{\text{calc}} = \frac{1}{N} \sum_{k=1}^N t_{\text{calc},k}, \tag{55}$$

where N is the maximum amount of steps $N = 100t_f$ due to a time step size $\Delta t = 10$ ms. To estimate measurement errors, the mean calculation time \bar{t}_{calc} is measured 10 times for each formulation. Its mean and twice the standard deviation is shown in Fig. 6 for the linear implicit Euler and the implicit Euler with at most $i_{\text{max}} = 10$ iterations for both RC3 and GC3. For comparison, the time necessary to evaluate the analytical solution \mathbf{u}_{flat} is shown as well. Due to possibly more Newton iterations, the implicit Euler schemes are at most 22% slower than the linear implicit Euler schemes. This difference is rather small because Newton’s method converges within 2 to 3 iterations for this experiment. The mean calculation time \bar{t}_{calc} of the servo-constraints formulation GC3 and RC3 is approximately 74% and 113% larger compared to the analytical solution, respectively. However, all mean calculation times are orders of magnitude smaller than the available control loop step time $\Delta t = 10$ ms, proving again the real-time capability. Therefore, online generation of feedforward inputs is possible. Moreover, there is enough time for far more complex systems to be solved in real-time.

Controller robustness is analyzed by introducing an initial trolley position error $s_{\Delta} = 0.5$ m to trajectory 1. For this experiment, the actual initial position is chosen as $\mathbf{p}_0 = [11 \ 0 \ 4.5]$ m and the final position is $\mathbf{p}_f = [15 \ 0 \ 6]$ m, while the transition time is $t_f = 15$ s. This is called trajectory 2. The measured load position $\hat{\mathbf{z}}$ and the input and state trajectories are shown in Fig. 7. The pure feedforward control SC cannot detect the initial error and finishes the trajectory with the same error s_{Δ} . The controllers which include a feedback loop detect the initial position error. The LQR controller provides a large initial input u_s to reduce the initial error quickly. This results in load swinging, which is reduced during the course of the trajectory. On the other hand, the SMC has a slower reaction to the initial error. Less load swinging is introduced. However, convergence to the desired trajectory is slower and the trajectory error $e(t)$ is larger compared to the LQR controller.

With the presented problem formulation, trajectory control is not only possible for straight but also for various other trajectories. For example, a semicircular trajectory can be used for collision avoidance. Experimental results are shown for a semicircle with radius $r = 0.95$ m in Fig. 8, where the initial position is $\mathbf{p}_0 = [9.65 \ 0 \ 4.5]$ m and the transition time $t_f = 9$ s. This is called trajectory 3. Figure 8(a) shows that all controllers are able to follow this trajectory. The sliding mode controller shows some performance drawbacks

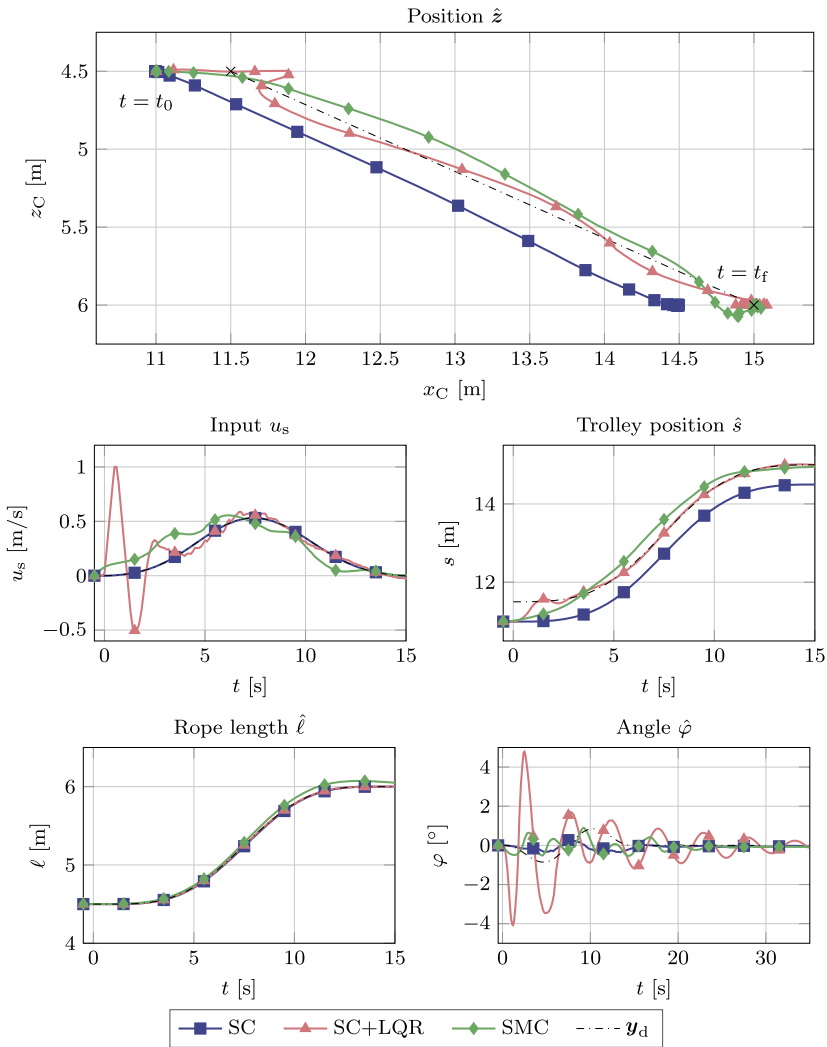


Fig. 7 Experimental results for the straight trajectory 2 with initial error $s_{\Delta} = 0.5$ m

for this trajectory. The feedforward controller augmented by LQR yields the best tracking results here. It reduces the small tracking errors initiated by the feedforward controller. An overlay of videos frames of the experiment is shown in Fig. 8(b).

In total, the sliding mode controller shows superior performance in load swing damping, but the combination of servo-constraints feedforward and LQR control yields better results in trajectory tracking.

7 Conclusion

Servo-constraints are applicable to underactuated multibody systems described in either generalized or redundant coordinates. For the presented system, the formulation in general-

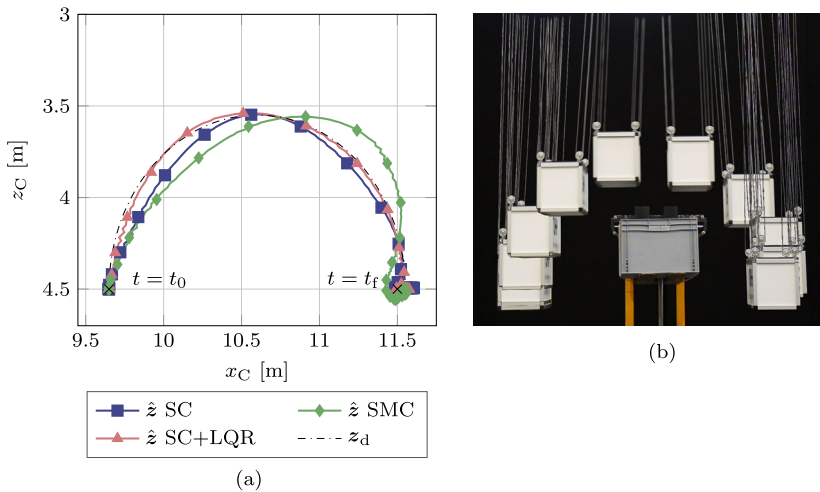


Fig. 8 Experimental results for tracking a semicircle with radius $r = 0.95$ m applicable for collision avoidance, trajectory 3. **(a)** Measured position \hat{z} . **(b)** Overlaid video frames of the experiment

ized coordinates yields an index 5 DAE which has to be reduced to index 3. By formulating the system dynamics in redundant coordinates, a more simple representation of the servo-constraints is possible. By directly applying these constraints to the equations of motion, an index 3 DAE system arises. No further projection is necessary. Servo-constraints pose a simple method to design a feedforward tracking controller. Experiments with an overhead crane test bench show that the servo-constraints formulation can be solved in real-time using a linear implicit Euler scheme. The servo-constraints solution can be augmented with an LQR feedback controller for a robust controller performance. Moreover, it can also be used as a basis for sliding mode control.

Compliance with ethical standards The authors declare that they have no conflict of interest.

References

1. Abdel-Rahman, E.M., Nayfeh, A.H., Masoud, Z.N.: Dynamics and control of cranes: a review. *J. Vib. Control* **9**(7), 863–908 (2003)
2. Ashrafiuon, H., Erwin, R.S.: Sliding mode control of underactuated multibody systems and its application to shape change control. *Int. J. Control* **81**(12), 1849–1858 (2008)
3. Betsch, P., Altmann, R., Yang, Y.: Numerical integration of underactuated mechanical systems subjected to mixed holonomic and servo constraints. In: *Multibody Dynamics*, pp. 1–18 (2016)
4. Blajer, W., Kołodziejczyk, K.: A geometric approach to solving problems of control constraints: theory and a DAE framework. *Multibody Syst. Dyn.* **11**(4), 343–364 (2004)
5. Blajer, W., Kołodziejczyk, K.: Improved DAE formulation for inverse dynamics simulation of cranes. *Multibody Syst. Dyn.* **25**(2), 131–143 (2011)
6. Boustany, F., D'Andrea-Novet, B.: Adaptive control of an overhead crane using dynamic feedback linearization and estimation design. In: *Robotics and Automation, 1992, Proceedings, 1992 IEEE International Conference*, pp. 1963–1968 (1992)
7. Brogan, W.L.: *Modern Control Theory*. Prentice Hall, New York (1991)
8. Campbell, S.L.: High-index differential algebraic equations. *Mech. Struct. Mach.* **23**(2), 199–222 (1995)
9. Castillo, I., Vazquez, C., Fridman, L.: Overhead crane control through LQ singular surface design Matlab toolbox. In: *American Control Conference (ACC), 2015*, pp. 5847–5852 (2015)

10. Fliess, M., Lévine, J., Martin, P., Rouchon, P.: Flatness and defect of non-linear systems: introductory theory and examples. *Int. J. Control* **61**(6), 1327–1361 (1995)
11. Hairer, E., Wanner, G.: *Stiff and Differential-Algebraic Problems*, 2nd edn. Springer, Berlin (1996)
12. Otto, S.: *Nonlinear trajectory control of a gantry crane*. Master thesis MSC-001, Institute of Mechanics and Ocean Engineering, Hamburg University of Technology (2016)
13. Schiehlen, W., Eberhard, P.: *Technische Dynamik Rechnergestützte Modellierung mechanischer Systeme im Maschinen- und Fahrzeugbau*, 4th edn. Springer, Wiesbaden (2014)
14. Seifried, R., Blajer, W.: Analysis of servo-constraint problems for underactuated multibody systems. *Mech. Sci.* **4**(1), 113–129 (2013)
15. Slotine, J.J.E., Li, W.: *Applied Nonlinear Control*. Prentice Hall International, Englewood Cliffs (1991)