8

Model Predictive Trajectory Control for High-Speed Rack Feeders

Harald Aschemann and Dominik Schindele
Chair of Mechatronics, University of Rostock
18059 Rostock, Germany

1. Introduction

Rack feeders represent the commonly used handling systems for the automated operation of high-bay rackings. To further increase the handling capacity by shorter transport times, control measures are necessary for the reduction of excited structural oscillations, see also Aschemann & Ritzke (2009). One possible approach is given by flatness-based feedforward control, where the desired control inputs are determined by dynamic system inversion using the desired trajectories for the flat outputs as in Bachmayer et al. (2008) and M. Bachmayer & Ulbrich (2008). However, both publications consider only a constant mass position in vertical direction on an elastic beam without any feedback control. A variational approach is presented in Kostin & Saurin (2006) to compute an optimal feedforward control for an elastic beam. Unfortunately, feedforward control alone is not sufficient to guarantee small tracking errors when model uncertainty is present or disturbances act on the system. For this reason in this contribution a model predictive control (MPC) design is presented for fast trajectory control. In general, at model predictive control the optimal input vector is mostly calculated by minimising a quadratic cost function as, e.g., in Wang & Boyd (2010) or Magni & Scattolini (2004). In contrast, the here considered MPC approach aims at reducing future state errors, see Jung & Wen (2004), and allows for a relatively small computational effort as required in a real-time implementation. Hence, the proposed MPC algorithm is well suited for systems with fast dynamics, e.g., a high-speed linear axis with pneumatic muscles as presented in Schindele & Aschemann (2008) or high-speed rack feeders as in the given case. A further attractive characteristic of this MPC approach is its applicability to linear as well as nonlinear systems.
For the experimental investigation of modern control approaches to active oscillation damping as well as tracking control, a test rig of a high-speed rack feeder has been build up at the Chair of Mechatronics at the University of Rostock, see Figure 1. The experimental set-up consists of a carriage driven by an electric DC servo motor via a toothed belt, on which an elastic beam as the vertical supporting structure is mounted. On this beam structure, a cage with variable load mass is guided relocatably in vertical direction. This cage with the coordinate $y_K(t)$ in horizontal direction and $x_K(t)$ in vertical direction represents the tool center point (TCP) of the rack feeder that should track desired trajectories as accurate as possible. The movable cage is driven by a tooth belt and an electric DC servo motor as well. The angles of the actuators are measured by internal angular transducers, respectively. Additionally, the horizontal position of the carriage is detected by a magnetostrictive transducer. Both axes are operated with a fast underlying velocity control on the current converter. Consequently, the

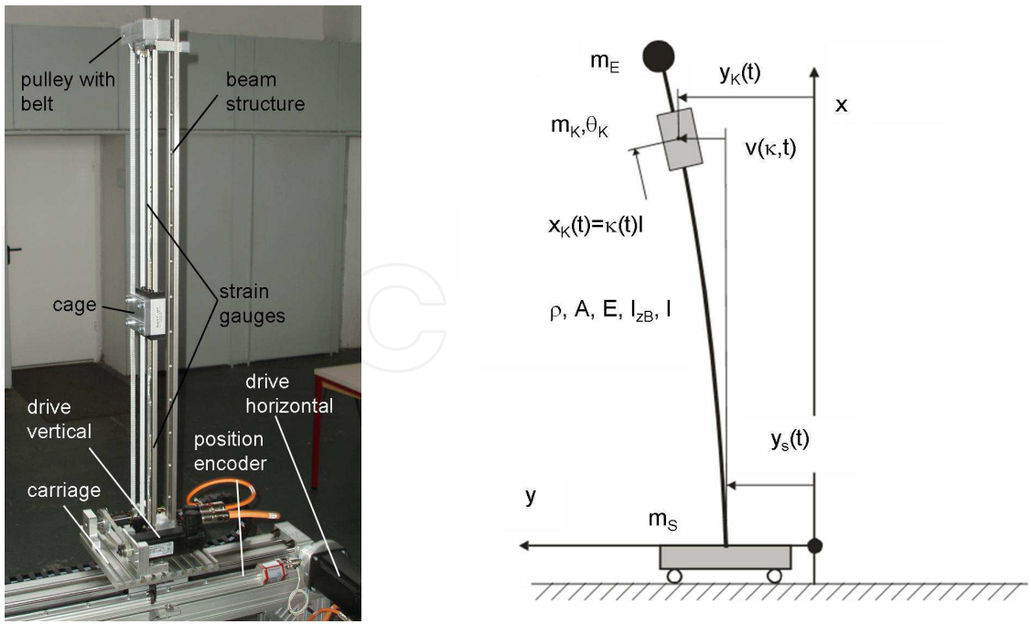

Fig. 1. Experimental set-up of the high-speed rack feeder (left) and the corresponding elastic multibody model (right).

corresponding velocities deal as new control input, and the implementational effort is tremendously reduced as compared to the commonly used force or torque input, like in Staudecker et al. (2008), where passivity techniques were employed for feedback control of a similar set-up. Two strain gauges are used to determine the bending deformation of the elastic beam.
Basis of the control design for the rack feeder is a planar elastic multibody system, where for the mathematical description of the bending deflection of the elastic beam a Ritz ansatz is introduced, covering for instance the first bending mode. The decentralised feedforward and feedback control design for both axes is performed employing a linearised state space representation, respectively. Given couplings between both axes are taken into account by the gain-scheduling technique with the normalised vertical cage position as scheduling parameter, see also Aschemann & Ritzke (2010). This leads to an adaptation of the whole control structure for the horizontal axis. The capability of the proposed control concept is shown by experimental results from the test set-up with regard to tracking behaviour and damping of bending oscillations. Especially the artificial damping introduced by the closed control loop represents a main improvement. The maximum velocity of the TCP during the tracking experiments is approx. 2.5 m/s.

2. Control-oriented modelling of the mechatronic system

Elastic multibody models have proven advantageously for the control-oriented modelling of flexible mechanical systems. For the feedforward and feedback control design of the rack feeder a multibody model with three rigid bodies - the carriage (mass m_S), the cage movable on the beam structure (mass m_K, mass moment of inertia θ_K), and the end mass at the tip of the beam (mass m_E) - and an elastic Bernoulli beam (density ρ, cross sectional area A, Youngs modulus E, second moment of area I_{zB}, and length ℓ) is chosen. The varying vertical position

$x_K(t)$ of the cage on the beam is denoted by the dimensionless system parameter

$$\kappa\left(t\right) = \frac{x_\kappa\left(t\right)}{l}. \tag{1}$$

The elastic degrees of freedom of the beam concerning the bending deflection can be described by the following Ritz ansatz

$$v\left(x,t\right) = \bar{\bar{v}}_1\left(x\right) v_1\left(t\right) = \left[\frac{3}{2}\left(\frac{x}{l}\right)^2 - \frac{1}{2}\left(\frac{x}{l}\right)^3\right] v_1\left(t\right), \tag{2}$$

which takes into account only the first bending mode. The vector of generalised coordinates results in

$$\boldsymbol{q}\left(t\right) = \left[\begin{array}{c} y_S\left(t\right) \\ v_1\left(t\right) \end{array}\right]. \tag{3}$$

The nonlinear equations of motion can be derived either by Lagrange's equations or, advantageously, by the Newton-Euler approach, cf. Shabana (2005). After a linearisation for small bending deflections, the equations of motion can be stated in M-D-K form

$$\boldsymbol{M}\ddot{\boldsymbol{q}}\left(t\right) + \boldsymbol{D}\dot{\boldsymbol{q}}\left(t\right) + \boldsymbol{K}\boldsymbol{q}\left(t\right) = \boldsymbol{h}\cdot\left[F_{SM}\left(t\right) - F_{SR}\left(\dot{y}_S\left(t\right)\right)\right]. \tag{4}$$

The symmetric mass matrix is given by

$$\boldsymbol{M} = \left[\begin{array}{cc} m_S + \rho Al + m_K + m_E & \frac{3}{8}\rho Al + \frac{m_K\kappa^2}{2}\left[3-\kappa\right] + m_E \\ \frac{3}{8}\rho Al + \frac{m_K\kappa^2}{2}\left[3-\kappa\right] + m_E & m_{22} \end{array}\right], \tag{5}$$

with $m_{22} = \frac{33}{140}\rho Al + \frac{6\rho I_{zB}}{5l} + \frac{m_K\kappa^2}{4}\left[3-\kappa\right]^2 + \frac{9\theta_K\kappa^2}{l^2}\left[1-\kappa+\frac{\kappa^2}{4}\right] + m_E$. The damping matrix, which is specified with stiffness-proportional damping properties, and the stiffness matrix become

$$\boldsymbol{D} = \left[\begin{array}{cc} 0 & 0 \\ 0 & \frac{3k_dEI_{zB}}{l^3} \end{array}\right], \tag{6}$$

$$\boldsymbol{K} = \left[\begin{array}{cc} 0 & 0 \\ 0 & \frac{3EI_{zB}}{l^3} - \frac{3}{8}\rho Ag - \frac{3m_Kg\kappa^3}{l}\left[1+\frac{3\kappa^2}{20}-\frac{3\kappa}{4}\right] - \frac{6m_Eg}{5l} \end{array}\right]. \tag{7}$$

The input vector of the generalised forces, which accounts for the control input as well as the disturbance input, reads

$$\boldsymbol{h} = \left[\begin{array}{cc} 1 & 0 \end{array}\right]^T. \tag{8}$$

The electric drive for the carriage is operated with a fast underlying velocity control on the current converter. The resulting dynamic behaviour is characterised by a first-order lag system with a time constant T_{1y}

$$T_{1y}\ddot{y}_S\left(t\right) + \dot{y}_S\left(t\right) = v_S\left(t\right). \tag{9}$$

This differential equation replaces now the equation of motion for the carriage in the mechanical system model, which leads to a modified mass matrix as well as a modified damping matrix

$$\boldsymbol{M}_y = \left[\begin{array}{cc} T_{1y} & 0 \\ \frac{3}{8}\rho Al + \frac{m_K\kappa^2}{2}\left[3-\kappa\right] + m_E & m_{22} \end{array}\right], \tag{10}$$

$$\boldsymbol{D}_y = \begin{bmatrix} 1 & 0 \\ 0 & \frac{3k_d E I_{zB}}{l^3} \end{bmatrix} . \tag{11}$$

The stiffness matrix $\boldsymbol{K} = \boldsymbol{K}_y$ and the input vector for the generalised forces $\boldsymbol{h} = \boldsymbol{h}_y$, however, remain unchanged. Hence, the equations of motion are given by

$$\ddot{\boldsymbol{q}} = -\boldsymbol{M}_y^{-1}\boldsymbol{K}_y\boldsymbol{q} - \boldsymbol{M}_y^{-1}\boldsymbol{D}_y\dot{\boldsymbol{q}} + \boldsymbol{M}_y^{-1}\boldsymbol{h}_y v_S . \tag{12}$$

For control design, the system representation is reformulated in state space form

$$\dot{\boldsymbol{x}}_y = \begin{bmatrix} \dot{\boldsymbol{q}} \\ \ddot{\boldsymbol{q}} \end{bmatrix} = \underbrace{\begin{bmatrix} \boldsymbol{0} & \boldsymbol{I} \\ -\boldsymbol{M}_y^{-1}\boldsymbol{K}_y & -\boldsymbol{M}_y^{-1}\boldsymbol{D}_y \end{bmatrix}}_{\boldsymbol{A}_y} \underbrace{\begin{bmatrix} \boldsymbol{q} \\ \dot{\boldsymbol{q}} \end{bmatrix}}_{\boldsymbol{x}_y} + \underbrace{\begin{bmatrix} \boldsymbol{0} \\ \boldsymbol{M}_y^{-1}\boldsymbol{h}_y \end{bmatrix}}_{\boldsymbol{b}_y} \underbrace{v_S}_{u_y} . \tag{13}$$

The design model for the vertical movement of the cage can be directly stated in state space representation. Here, an underlying velocity control is employed on the current converter, which is also described by a first-order lag system

$$T_{1x}\ddot{x}_K(t) + \dot{x}_K(t) = v_K(t) . \tag{14}$$

The corresponding state space description follows immediately in the form

$$\dot{\boldsymbol{x}}_x = \begin{bmatrix} \dot{x}_K \\ \ddot{x}_K \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 1 \\ 0 & -\frac{1}{T_{1x}} \end{bmatrix}}_{\boldsymbol{A}_x} \underbrace{\begin{bmatrix} x_K \\ \dot{x}_K \end{bmatrix}}_{\boldsymbol{x}_x} + \underbrace{\begin{bmatrix} 0 \\ \frac{1}{T_{1x}} \end{bmatrix}}_{\boldsymbol{b}_x} \underbrace{v_K}_{u_x} . \tag{15}$$

Whereas the state space respresentation for the horizontal y-axis depends on the varying system parameter $\kappa(t)$, the description of the x-axis is invariant. A gain-scheduling, hence, is necessary only for the horizontal axis in y-direction.

3. Decentralised control design

As for control, a decentralised approach is followed, at which the coupling of the vertical cage motion with the horizontal axis is taken into account by gain-scheduling techniques. For the control of the cage position $x_K(t)$ a simple proportional feedback in combination with feedforward control, which is based on the inverse transfer function of this axis, is sufficient

$$v_K(t) = K_R\left(x_{Kd}(t) - x_K(t)\right) + \dot{x}_{Kd}(t) + T_{1x}\ddot{x}_{Kd}(t) . \tag{16}$$

For this purpose, the desired trajectory $x_{Kd}(t)$ and its first two time derivatives are available from trajectory planning. The design of the state feedback for the horizontal motion is carried out by the MPC approach, which is explained in the following chapter.

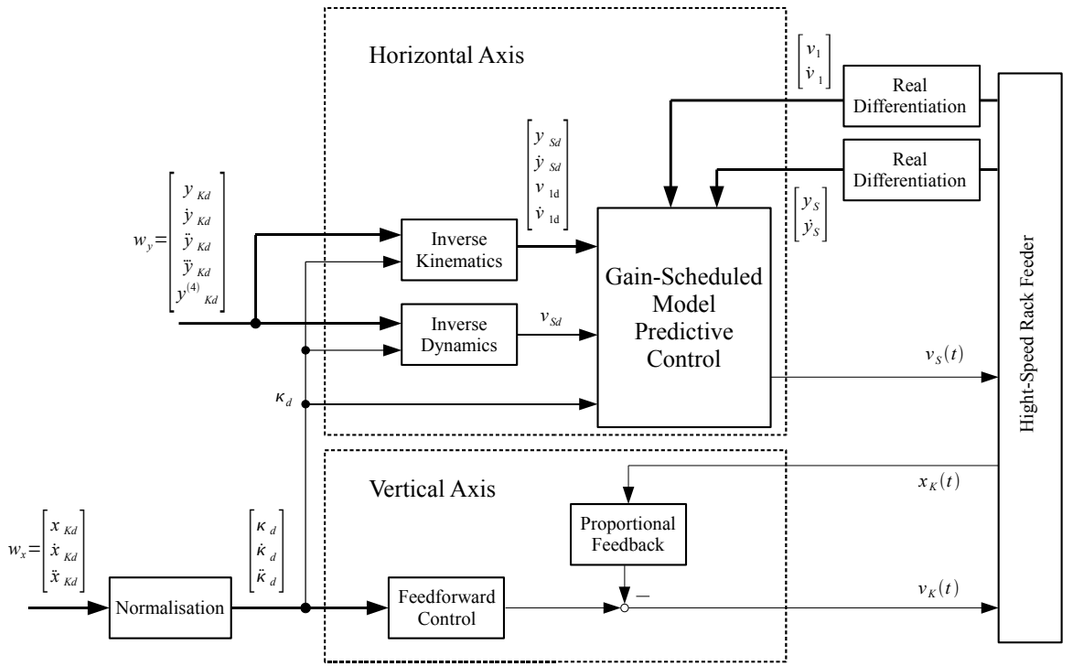

Fig. 2. Implementation of the control structure.

4. Model Predictive Control

The main idea of the control approach consists in a minimisation of a future tracking error in terms of the predicted state vector based on the actual state and the desired state vector resulting from trajectory planning, see Lizarralde et al. (1999), Jung & Wen (2004). The minimisation is achieved by repeated approximate numerical optimisation in each time step, in the given case using the Newton-Raphson technique. The optimisation is initialised in each time step with the optimisation result of the preceding time step in form of the input vector. The MPC-algorithm is based on the following discrete-time state space representation

$$\boldsymbol{x}_{k+1} = \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{b}u_k \,, \tag{17}$$

$$y_k = \boldsymbol{c}^T\boldsymbol{x}_k \,, \tag{18}$$

with the state vector $\boldsymbol{x}_k \in \mathbb{R}^n$, the control input $u_k \in \mathbb{R}$ and the output vector $y_k \in \mathbb{R}$. The constant M specifies the prediction horizon T_P as a multiple of the sampling time t_s, i.e. $T_P = M \cdot t_s$. The predicted input vector at time k becomes

$$\boldsymbol{u}_{k,M} = \left[u_1^{(k)}, ..., u_M^{(k)}\right]^T \,, \tag{19}$$

with $\boldsymbol{u}_{k,M} \in \mathbb{R}^M$. The predicted state vector at the end of the prediction horizon $\boldsymbol{\phi}_M(\boldsymbol{x}_k, \boldsymbol{u}_{k,M})$ is obtained by repeated substitution of k by $k+1$ in the discrete-time state equation (17)

$$\begin{aligned} \boldsymbol{x}_{k+2} &= \boldsymbol{A}\boldsymbol{x}_{k+1} + \boldsymbol{b}u_{k+1} = \boldsymbol{A}^2\boldsymbol{x}_k + \boldsymbol{A}\boldsymbol{b}u_k + \boldsymbol{b}u_{k+1} \\ &\vdots \\ \boldsymbol{x}_{k+M} &= \boldsymbol{A}^M\boldsymbol{x}_k + \boldsymbol{A}^{M-1}\boldsymbol{b}u_k + \boldsymbol{A}^{M-2}\boldsymbol{b}u_{k-1} + \ldots + \boldsymbol{b}u_{k+M-1} \\ &= \boldsymbol{\phi}_M(\boldsymbol{x}_k, \boldsymbol{u}_{k,M}) \, . \end{aligned} \tag{20}$$

The difference of $\boldsymbol{\phi}_M(\boldsymbol{x}_k, \boldsymbol{u}_{k,M})$ and the desired state vector $\boldsymbol{x}_d$ yields the final control error

$$\boldsymbol{e}_{M,k} = \boldsymbol{\phi}_M(\boldsymbol{x}_k, \boldsymbol{u}_{k,M}) - \boldsymbol{x}_d \, , \tag{21}$$

i.e. to the control error at the end of the prediction horizon. The cost function to be minimised follows as

$$J_{MPC} = \frac{1}{2} \cdot \boldsymbol{e}_{M,k}^T \boldsymbol{e}_{M,k} \, , \tag{22}$$

and, hence, the necessary condition for an extremum can be stated as

$$\frac{\partial J_{MPC}}{\partial \boldsymbol{e}_{M,k}} = \boldsymbol{e}_{M,k} \stackrel{!}{=} 0 \, . \tag{23}$$

A Taylor-series expansion of (23) at $\boldsymbol{u}_{k,M}$ in the neighbourhood of the optimal solution leads to the following system of equations

$$\boldsymbol{0} = \boldsymbol{e}_{M,k} + \frac{\partial \boldsymbol{\phi}_M}{\partial \boldsymbol{u}_{k,M}} \Delta \boldsymbol{u}_{k,M} + T.h.O.. \tag{24}$$

The vector $\Delta \boldsymbol{u}_{k,M}$ denotes the difference which has to be added to the input vector $\boldsymbol{u}_{k,M}$ to obtain the optimal solution. The n equations (24) represent an under-determined set of equations with $m \cdot M$ unknowns having an infinite number of solutions. An unique solution for $\Delta \boldsymbol{u}_{k,M}$ can be determined by solving the following L_2-optimisation problem with (24) as side condition

$$J = \frac{1}{2} \cdot \Delta \boldsymbol{u}_{k,M}^T \Delta \boldsymbol{u}_{k,M} + \boldsymbol{\lambda}^T \left(\boldsymbol{e}_{M,k} + \frac{\partial \boldsymbol{\phi}_M}{\partial \boldsymbol{u}_{k,M}} \Delta \boldsymbol{u}_{k,M} \right) \, . \tag{25}$$

Consequently, the necessary conditions can be stated as

$$\begin{aligned} \frac{\partial J}{\partial \Delta \boldsymbol{u}_{k,M}} \stackrel{!}{=} 0 &= \Delta \boldsymbol{u}_{k,M} + \left(\frac{\partial \boldsymbol{\phi}_M}{\partial \boldsymbol{u}_{k,M}} \right)^T \boldsymbol{\lambda}, \\ \frac{\partial J}{\partial \boldsymbol{\lambda}} \stackrel{!}{=} 0 &= \boldsymbol{e}_{M,k} + \frac{\partial \boldsymbol{\phi}_M}{\partial \boldsymbol{u}_{k,M}} \Delta \boldsymbol{u}_{k,M}, \end{aligned} \tag{26}$$

which results in $\boldsymbol{e}_{M,k}$

$$\boldsymbol{e}_{M,k} = \underbrace{\frac{\partial \boldsymbol{\phi}_M}{\partial \boldsymbol{u}_{k,M}} \left(\frac{\partial \boldsymbol{\phi}_M}{\boldsymbol{u}_{k,M}} \right)^T}_{\boldsymbol{S}(\boldsymbol{\phi}_M, \boldsymbol{u}_{k,M})} \boldsymbol{\lambda} \, . \tag{27}$$

If the matrix $\boldsymbol{S}\left(\boldsymbol{\phi}_M, \boldsymbol{u}_{k,M}\right)$ is invertible, the vector $\boldsymbol{\lambda}$ can be calculated as follows

$$\boldsymbol{\lambda} = \boldsymbol{S}^{-1}\left(\boldsymbol{\phi}_M, \boldsymbol{u}_{k,M}\right) \boldsymbol{e}_{M,k} \, . \tag{28}$$

An almost singular matrix $\boldsymbol{S}\left(\boldsymbol{\phi}_M, \boldsymbol{u}_{k,M}\right)$ can be treated by a modification of (28)

$$\boldsymbol{\lambda} = \left[\mu \boldsymbol{I} + \boldsymbol{S}\left(\boldsymbol{\phi}_M, \boldsymbol{u}_{k,M}\right)\right]^{-1} \boldsymbol{e}_{M,k} \,, \tag{29}$$

where $\boldsymbol{I}$ denotes the unity matrix. The regularisation parameter $\mu > 0$ in (29) may be chosen constant or may be calculated by a sophisticated algorithm. The latter solution improves the convergence of the optimisation but increases, however, the computational complexity. Solving (26) for $\Delta \boldsymbol{u}_{k,M}$ and inserting $\boldsymbol{\lambda}$ according to (28) or (29), directly yields the L_2-optimal solution

$$\Delta \boldsymbol{u}_{k,M} = -\left(\frac{\partial \boldsymbol{\phi}_M}{\partial \boldsymbol{u}_{k,M}}\right)^T \boldsymbol{S}^{-1}\left(\boldsymbol{\phi}_M, \boldsymbol{u}_{k,M}\right) \boldsymbol{e}_{M,k} = -\left(\frac{\partial \boldsymbol{\phi}_M}{\partial \boldsymbol{u}_{k,M}}\right)^{\dagger} \boldsymbol{e}_{M,k} \,. \tag{30}$$

Here, $\left(\frac{\partial \boldsymbol{\phi}_M}{\partial \boldsymbol{u}_{k,M}}\right)^{\dagger}$ denotes the Moore-Penrose pseudo inverse of $\frac{\partial \boldsymbol{\phi}_M}{\partial \boldsymbol{u}_{k,M}}$. The overall MPC-algorithm can be described as follows:
Choice of the initial input vector $\boldsymbol{u}_{0,M}$ at time $k = 0$, e.g. $\boldsymbol{u}_{0,M} = \boldsymbol{0}$, and repetition of steps a) - c) at each sampling time $k \geq 0$:

a) Calculation of an improved input vector $\boldsymbol{v}_{k,M}$ according to

$$\boldsymbol{v}_{k,M} = \boldsymbol{u}_{k,M} - \eta_k \left(\frac{\partial \boldsymbol{\phi}_M}{\partial \boldsymbol{u}_{k,M}}\right)^{\dagger} \boldsymbol{e}_{M,k} \,. \tag{31}$$

The step width η_k can be determined with, e.g., the Armijo-rule.

b) For the calculation of $\boldsymbol{u}_{k+1,M}$ the elements of the vector $\boldsymbol{v}_{k,M}$ have to be shifted by one element and the steady-state input u_d corresponding to the final state has to be inserted at the end

$$\boldsymbol{u}_{k+1,M} = \begin{bmatrix} \boldsymbol{0}_{((M-1)\times 1)} \\ 1 \end{bmatrix} u_d + \begin{bmatrix} \boldsymbol{0}_{((M-1)\times 1)} & \boldsymbol{I}_{(M-1)} \\ 0 & \boldsymbol{0}_{(1\times(M-1))} \end{bmatrix} \boldsymbol{v}_{k,M}. \tag{32}$$

In general, the steady-state control input u_d can be computed from

$$\boldsymbol{x}_d = \boldsymbol{A}\boldsymbol{x}_d + \boldsymbol{b}u_d. \tag{33}$$

Alternatively, the desired input vector u_d can be calculated by an inverse system model. If the system is differentially flat, see Fliess et al. (1995) the desired input u_d can be calculated exactly by the flat system output and a finite number of its time derivative. For non-flat outputs -as in the given case- the approach presented in chapter 4.4 is useful.

c) The first element of the improved input vector $\boldsymbol{v}_{k,M}$ is applied as control input at time k

$$u_k = \begin{bmatrix} 1 & \boldsymbol{0}_{(1\times(M-1))} \end{bmatrix} \boldsymbol{v}_{k,M} \,. \tag{34}$$

In the proposed algorithm only one iteration is performed per time step. A similar approach using several iteration steps is described in Weidemann et al. (2004). An improvement of the trajectory tracking behaviour can be achieved if an input vector resulting from an inverse system model is used as initial vector for the subsequent optimisation step instead of the last input vector. The slightly modified algorithm can be stated as follows

a) Calculation of the ideal input vector $\boldsymbol{u}_{k,M}^{(d)}$ by evaluating an inverse system model with the specified reference trajectory as well as a certain number $\beta \in \mathbb{N}$ of its time derivatives

$$\boldsymbol{u}_{k,M}^{(d)} = \boldsymbol{u}_{k,M}^{(d)} \left(\boldsymbol{y}_d, \dot{\boldsymbol{y}}_d, ..., \overset{(\beta)}{\boldsymbol{y}}_d \right) . \tag{35}$$

b) Calculation of the improved input vector $\boldsymbol{v}_{k,M}$ based on the equation

$$\boldsymbol{v}_{k,M} = \boldsymbol{u}_{k,M}^{(d)} - \eta_k \left(\frac{\partial \boldsymbol{\phi}_M}{\partial \boldsymbol{u}_{k,M}} \right)^{\dagger} \boldsymbol{e}_{M,k} . \tag{36}$$

c) Application of the first element of $\boldsymbol{v}_{k,M}$ to the process

$$u_k = \begin{bmatrix} 1 & \mathbf{0}_{(1\times(M-1))} \end{bmatrix} \boldsymbol{v}_{k,M} . \tag{37}$$

If the reference trajectory is known in advance, the according reference input vector $\boldsymbol{u}_{k,M}^{(d)}$ can be computed offline. Consequently, the online computational time remains unaffected.

4.1 Numerical calculations

The analytical computation of the Jacobian $\frac{\partial \boldsymbol{\phi}_M}{\partial \boldsymbol{u}_{k,M}}$ becomes increasingly complex for larger values of M. Therefore, a numerical approach is preferred taking advantage of the chain rule with $i = 0, ..., M-1$

$$\frac{\partial \boldsymbol{\phi}_M}{\partial \boldsymbol{u}_{i+1}^{(k)}} = \frac{\partial \boldsymbol{\phi}_M}{\partial \boldsymbol{x}_{k+M-1}} \cdot \frac{\partial \boldsymbol{x}_{k+M-1}}{\partial \boldsymbol{x}_{k+M-2}} \cdot \ldots \cdot \frac{\partial \boldsymbol{x}_{k+i+2}}{\partial \boldsymbol{x}_{k+i+1}} \cdot \frac{\partial \boldsymbol{x}_{k+i+1}}{\partial u_{i+1}^{(k)}} . \tag{38}$$

In this way, the Jacobian can be computed as follows

$$\frac{\partial \boldsymbol{\phi}_M}{\partial \boldsymbol{u}_{k,M}} = [\boldsymbol{A}^{M-1}\boldsymbol{b}, \boldsymbol{A}^{M-2}\boldsymbol{b}, ..., \boldsymbol{A}\boldsymbol{b}, \boldsymbol{b}] . \tag{39}$$

For the inversion of the symmetric and positive definite matrix $\boldsymbol{S}\left(\boldsymbol{\phi}_M, \boldsymbol{u}_{k,M}\right) = \frac{\partial \boldsymbol{\phi}_M}{\partial \boldsymbol{u}_{k,M}} \left(\frac{\partial \boldsymbol{\phi}_M}{\partial \boldsymbol{u}_{k,M}} \right)^T$ the Cholesky-decomposition has proved advantageous in terms of computational effort.

4.2 Choice of the MPC design parameters

The most important MPC design parameter is the prediction horizon T_P, which is given as the product of the sampling time t_s and the constant value M. Large values of T_P lead to a slow and smooth transient behaviour and result in a robust and stable control loop. For fast trajectory tracking, however, a smaller value T_P is desirable concerning a small tracking error. The choice of the sampling time t_s is crucial as well: a small sampling time is necessary regarding both discretisation error and stability; however, the MPC-algorithm has to be evaluated in real-time within the sampling inverval. Furthermore, the smaller t_s, the larger becomes M for a given prediction horizon, which in turn increases the computational complexity of the optimisation step. Consequently, a system-specific trade-off has to be made for the choice of M and t_s. This paper follows the moving horizon approach with a constant prediction horizon and, hence, a constant dimension $m \cdot M$ of the corresponding optimisation problem in constrast to the shrinking horizon approach according to Weidemann et al. (2004).

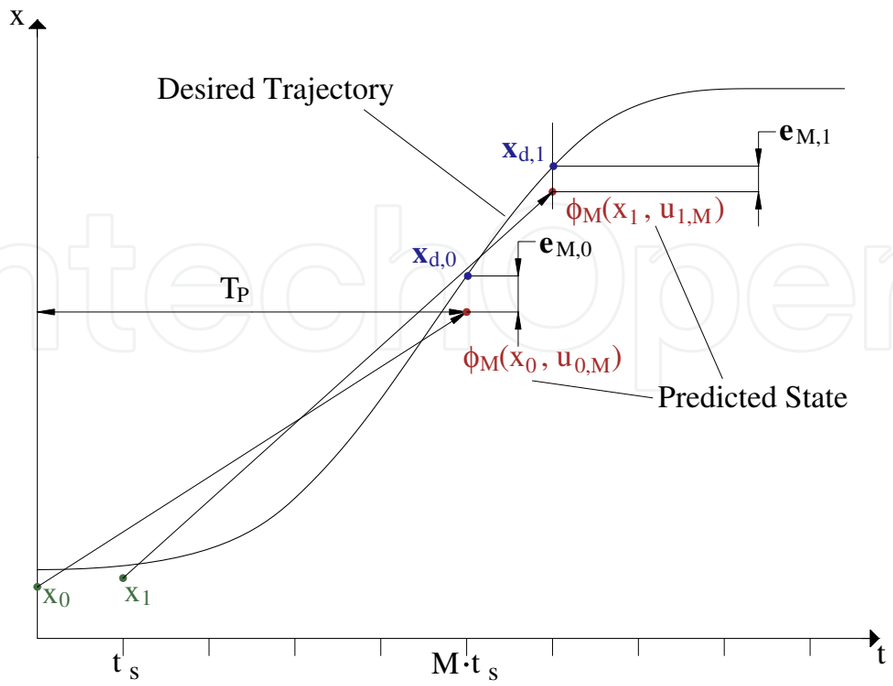

Fig. 3. Design parameters.

4.3 Input constraints

One major advantage of predictive control is the possibility to easily account for input constraints, which are present in almost all control applications. To this end, the cost function can be extended with a corresponding term

$$h(u_j^{(k)}) = \begin{cases} 0 & u_{min} \leq u_j^{(k)} \leq u_{max} \\ g_1(u_j^{(k)}) \text{ for } & u_j^{(k)} > u_{max} \\ g_2(u_j^{(k)}) & u_j^{(k)} < u_{min} \end{cases}, \tag{40}$$

which has to be evaluated componentwise, i.e. for each input variable at each sampling time. Thus, the contribution of the additional input constraints depending on $\boldsymbol{u}_{k,M}$ is given by

$$z(\boldsymbol{u}_{k,M}) = \sum_{j=1}^{M} h(u_j^{(k)}). \tag{41}$$

Instead of $\boldsymbol{e}_{M,k}$, the extended vector $\left[\boldsymbol{e}_{M,k}^T, z\right]^T$ has to be minimised in the MPC-algorithm.

4.4 MPC of the horizontal cage position

The state space representation for the cage position control in y-direction design is given by (13). The discrete-time representation of the continous-time system is obtained by Euler discretisation

$$\boldsymbol{x}_{y,k+1} = \left(\boldsymbol{I} + t_s \cdot \boldsymbol{A}_y(\kappa)\right) \boldsymbol{x}_{y,k} + t_s \cdot \boldsymbol{b}_y(\kappa)\, u_{y,k}. \tag{42}$$

Using this simple discretisation method, the computational effort for the MPC-algorithm can be kept acceptable. By the way, no significant improvement could be obtained for the given system with the Heun discretisation method because of the small sampling time $t_s = 3\ ms$. Only in the case of large sampling times, e.g. $t_s > 20\ ms$, the increased computational effort caused by a sophisticated time discretisation method is advantageous. Then, the smaller discretisation error allows for less time integration steps for a specified prediction horizon, i.e. a smaller number M. As a result, the smaller number of time steps can overcompensate the larger effort necessary for a single time step.
The ideal input $u_d(t)$ can be obtained in continous time as function of the output variable

$$y_K(t) = \boldsymbol{c}_y^T \boldsymbol{x}_y(t) = \begin{bmatrix} 1 & \frac{1}{2}\kappa^2\,(3-\kappa) & 0 & 0 \end{bmatrix} \boldsymbol{x}_y(t)\,, \tag{43}$$

and a certain number of its time derivatives. For this purpose the corresponding transfer function of the system under consideration is employed

$$\frac{Y_K\,(s)}{U_d\,(s)} = \boldsymbol{c}_y^T \left(s\boldsymbol{I} - \boldsymbol{A}_y\right)^{-1} \boldsymbol{b}_y = \frac{\left(b_0 + b_1 \cdot s + b_2 \cdot s^2\right)}{N\,(s)}\,. \tag{44}$$

Obviously, the numerator of the control transfer function contains a second degree polynomial in s, leading to two transfer zeros. This shows that the considered output $y_K(t)$ represents a non-flat output variable that makes computing of the feedforward term more difficult. A possible way for calculating the desired input variable is given by a modification of the numerator of the control transfer function by introducing a polynomial ansatz for the feedforward action according to

$$U_d\,(s) = \left[k_{V0} + k_{V1} \cdot s + \ldots + k_{V4} \cdot s^4\right] Y_{Kd}\,(s)\,. \tag{45}$$

For its realisation the desired trajectory $y_{Kd}(t)$ as well as the first four time derivatives are available from a trajectory planning module. The feedforward gains can be computed from a comparison of the corresponding coefficients in the numerator as well as the denominator polynomials of

$$\begin{aligned} \frac{Y_K\,(s)}{Y_{Kd}\,(s)} &= \frac{\left(b_0 + \ldots + b_2 \cdot s^2\right)\left[k_{V0} + \ldots + k_{V4} \cdot s^4\right]}{N\,(s)} \\ &= \frac{b_{V0}\left(k_{Vj}\right) + b_{V1}\left(k_{Vj}\right) \cdot s + \ldots + b_{V6}\left(k_{Vj}\right) \cdot s^6}{a_0 + a_1 \cdot s + \ldots + s^4} \end{aligned} \tag{46}$$

according to

$$a_i = b_{Vi}\left(k_{Vj}\right), i = 0, \ldots, n = 4\,. \tag{47}$$

This leads to parameter-dependent feedforward gains $k_{Vj} = k_{Vj}(\kappa)$. It is obvious that due the higher numerator degree in the modified control transfer function a remaining dynamics must be accepted. Lastly, the desired input variable in the time domain is represented by

$$u_d(t) = u_d\left(\dot{y}_{Kd}(t), \ddot{y}_{Kd}(t), \dddot{y}_{Kd}(t), y_{Kd}^{(4)}(t), \kappa\right)\,. \tag{48}$$

To obtain the desired system states as function of the output trajectory the output equation

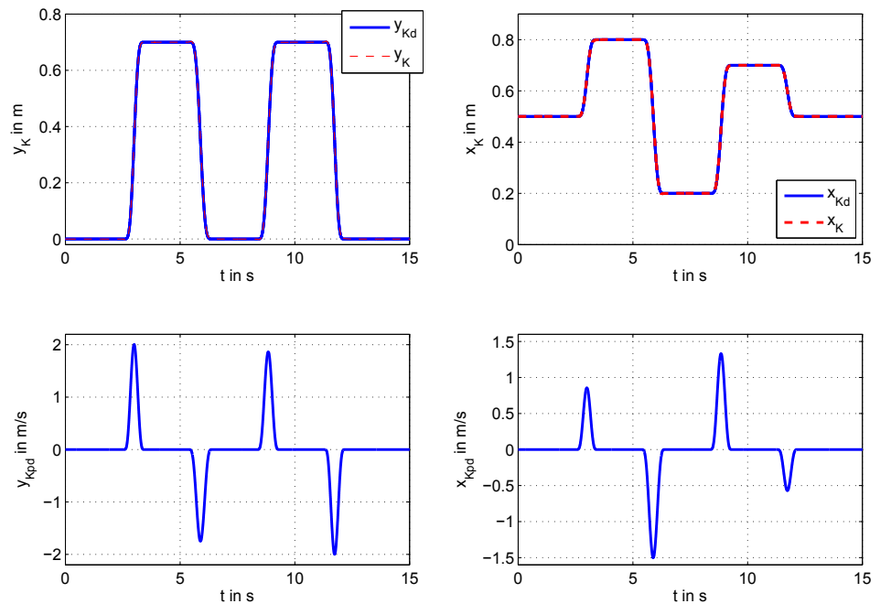

Fig. 4. Desired trajectories for the cage motion: desired and actual position in horizontal direction (upper left corner), desired and actual position in vertical direction (upper right corner), actual velocity in horizontal direction (lower left corner) and actual velocity in vertical direction (lower right corner).

and its first three time derivatives are considered. Including the equations of motion (12) yields the following set of equations

$$y_{Kd}(t) = y_S(t) + \tfrac{1}{2}\kappa^2(3-\kappa)\cdot v_1(t), \tag{49}$$

$$\dot{y}_{Kd}(t) = \dot{y}_S(t) + \tfrac{1}{2}\kappa^2(3-\kappa)\cdot \dot{v}_1(t), \tag{50}$$

$$\ddot{y}_{Kd}(t) = \ddot{y}_S(t) + \tfrac{1}{2}\kappa^2(3-\kappa)\cdot \ddot{v}_1(t) = \ddot{y}_K\left(v_1(t), \dot{y}_S(t), \dot{v}_1(t), u_d(t), \kappa\right), \tag{51}$$

$$\dddot{y}_{Kd}(t) = \dddot{y}_K\left(v_1(t), \dot{y}_S(t), \dot{v}_1(t), u_d(t), \dot{u}_d(t), \kappa\right). \tag{52}$$

Solving equation (49) to (52) for the system states results in the desired state vector

$$\boldsymbol{x}_d(t) = \begin{bmatrix} y_{Sd}\left(y_{Kd}(t), \dot{y}_{Kd}(t), \ddot{y}_{Kd}(t), \dddot{y}_{Kd}(t), u_d(t), \dot{u}_d(t), \kappa\right) \\ v_{1d}\left(\dot{y}_{Kd}(t), \ddot{y}_{Kd}(t), \dddot{y}_{Kd}(t), u_d(t), \dot{u}_d(t), \kappa\right) \\ \dot{y}_{Sd}\left(\dot{y}_{Kd}(t), \ddot{y}_{Kd}(t), \dddot{y}_{Kd}(t), u_d(t), \dot{u}_d(t), \kappa\right) \\ \dot{v}_{1d}\left(\dot{y}_{Kd}(t), \ddot{y}_{Kd}(t), \dddot{y}_{Kd}(t), u_d(t), \dot{u}_d(t), \kappa\right) \end{bmatrix}. \tag{53}$$

This equation still contains the inverse dynamics $u_d(t)$ and its time derivative $\dot{u}_d$. Substituting u_d for equation (48) and $\dot{u}_d(t)$ for the time derivative of (48), which can be calculated analyti-

cally, finally leads to

$$\boldsymbol{x}_d(t) = \begin{bmatrix} y_{Sd}\left(y_{Kd}(t), \dot{y}_{Kd}(t), \ddot{y}_{Kd}(t), \dddot{y}_{Kd}(t), y_{Kd}^{(4)}(t), y_{Kd}^{(5)}(t), \kappa\right) \\ v_{1d}\left(y_{Kd}(t), \dot{y}_{Kd}(t), \ddot{y}_{Kd}(t), \dddot{y}_{Kd}(t), y_{Kd}^{(4)}(t), y_{Kd}^{(5)}(t), \kappa\right) \\ \dot{y}_{Sd}\left(y_{Kd}(t), \dot{y}_{Kd}(t), \ddot{y}_{Kd}(t), \dddot{y}_{Kd}(t), y_{Kd}^{(4)}(t), y_{Kd}^{(5)}(t), \kappa\right) \\ \dot{v}_{1d}\left(y_{Kd}(t), \dot{y}_{Kd}(t), \ddot{y}_{Kd}(t), \dddot{y}_{Kd}(t), y_{Kd}^{(4)}(t), y_{Kd}^{(5)}(t), \kappa\right) \end{bmatrix}. \tag{54}$$

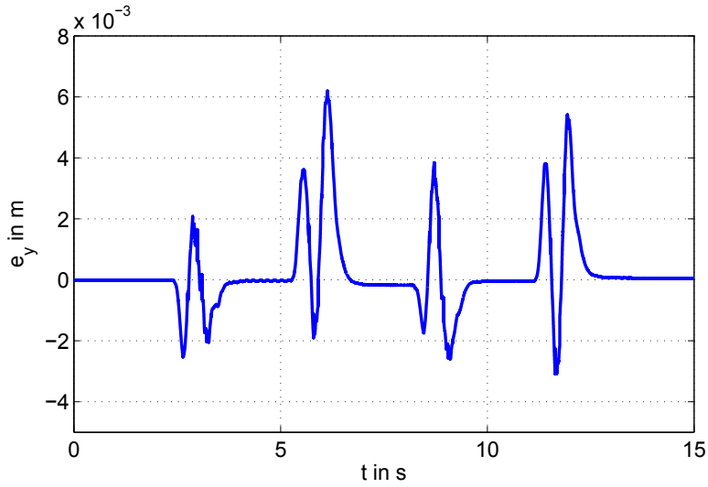

Fig. 5. Tracking error $e_y(t)$ for the cage motion in horizontal direction.

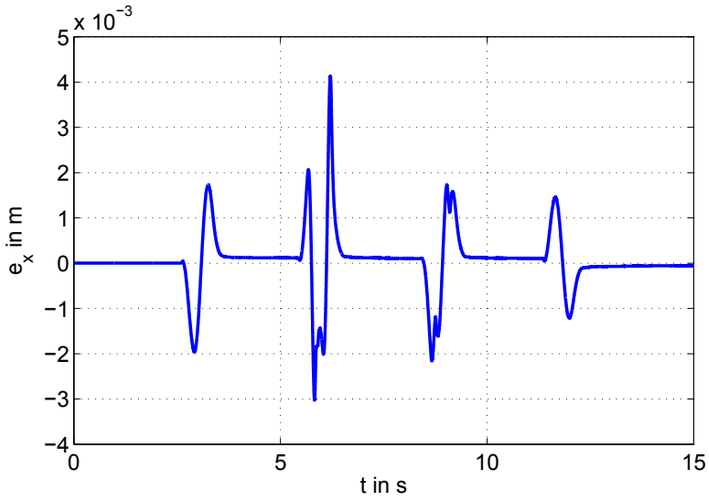

Fig. 6. Tracking error $e_x(t)$ for the cage motion in vertical direction.

5. Experimental validation on the test rig

The benefits and the efficiency of the proposed control measures shall be pointed out by experimental results obtained from the test set-up available at the Chair of Mechatronics, University of Rostock. For this purpose, a synchronous four times continuously differentiable desired trajectory is considered for the position of the cage in both x- and y-direction. The desired trajectory is given by polynomial functions that comply with specified kinematic constraints, which is achieved by taking advantage of time scaling techniques. The desired trajectory shown in Figure 4 comprises a sequence of reciprocating motions with maximum velocities of 2 m/s in horizontal direction and 1.5 m/s in vertical direction. The resulting tracking errors

$$e_y(t) = y_{Kd}(t) - y_K(t) \tag{55}$$

and

$$e_x(t) = x_{Kd}(t) - x_K(t) \tag{56}$$

are depicted in Figure 5 and Figure 6. As can be seen, the maximum position error in y-direction during the movements is about 6 mm and the steady-state position error is smaller than 0.2 mm, whereas the maximum position error in x-direction is approx. 4 mm. Figure 7

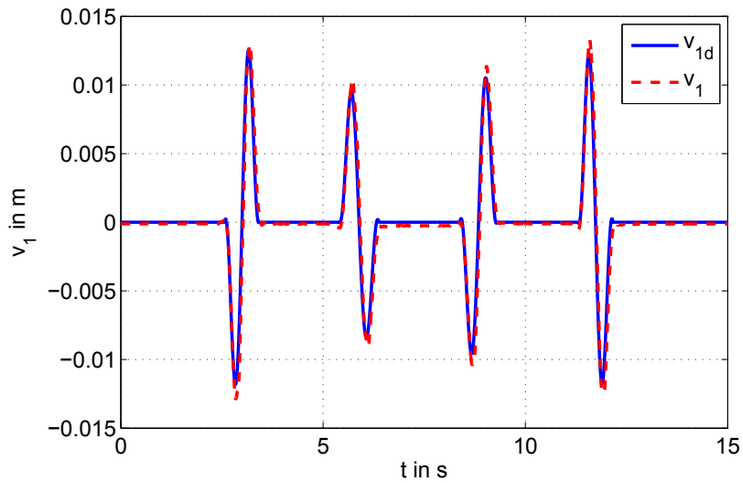

Fig. 7. Comparison of the desired values $v_{1d}(t)$ and the actual values $v_1(t)$ for the bending deflection.

shows the comparison of the bending deflection measured by strain gauges attached to the flexible beam with desired values. During the acceleration as well as the deceleration intervals, physically unavoidable bending deflections could be noticed. The achieved benefit is given by the fact the remaining oscillatons are negligible when the rack feeder arrives at its target position. This underlines both the high model accuracy and the quality of the active damping of the first bending mode. Figure 8 depicts the disturbance rejection properties due to an external excitation by hand. At the beginning, the control structure is deactivated, and the excited bending oscillations decay only due to the very weak material damping. After approx. 2.8 seconds, the control structure is activated and, hence, the first bending mode is actively damped. The remaining oscillations are characterised by higher bending modes that decay with material damping. In future work, the number of Ritz ansatz functions shall be

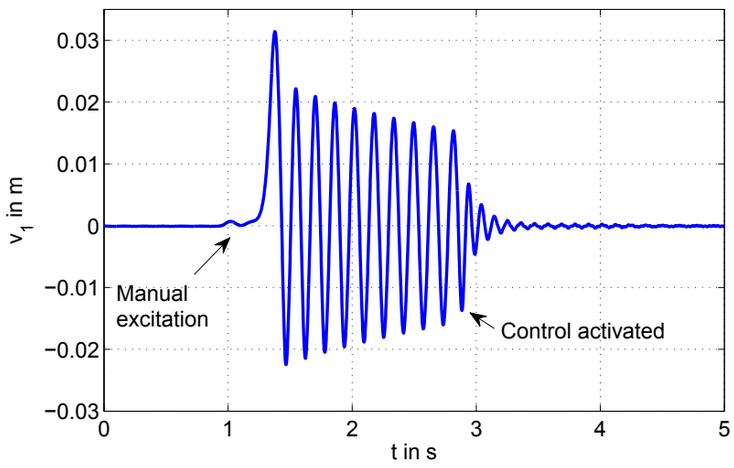

Fig. 8. Transient response after a manual excitation of the bending deflection: at first without feedback control, after approx. 2.8 seconds with active control.

increased to include the higher bending modes as well in the active damping. The corresponding elastic coordinates and their time derivatives can be determined by observer techniques.

6. Conclusions

In this paper, a gain-scheduled fast model predictive control strategy for high-speed rack feeders is presented. The control design is based on a control-oriented elastic multibody system. The suggested control algorithm aims at reducing the future tracking error at the end of the prediction horizon. Beneath an active oscillation damping of the first bending mode, an accurate trajectory tracking for the cage position in x- and y-direction is achieved. Experimental results from a prototypic test set-up point out the benefits of the proposed control structure. Experimental results show maximum tracking errors of approx. 6 mm in transient phases, whereas the steady-state tracking error is approx. 0.2 mm. Future work will address an active oscillation damping of higher bending modes as well as an additional gain-scheduling with respect to the varying payload.

7. References

Aschemann, H. & Ritzke, J. (2009). Adaptive aktive Schwingungsdämpfung und Trajektorienfolgeregelung für hochdynamische Regalbediengeräte (in German), *Schwingungen in Antrieben, Vorträge der 6. VDI-Fachtagung in Leonberg, Germany*. (in German).

Aschemann, H. & Ritzke, J. (2010). Gain-scheduled tracking control for high-speed rack feeders, *Proc. of the first joint international conference on multibody system dynamics (IMSD), 2010, Lappeenranta, Finland* .

Bachmayer, M., Rudolph, J. & Ulbrich, H. (2008). Flatness based feed forward control for a horizontally moving beam with a point mass, *European Conference on Structural Control, St. Petersburg* pp. 74–81.

Fliess, M., Levine, J., Martin, P. & Rouchon, P. (1995). Flatness and defect of nonlinear systems: Introductory theory and examples, *Int. J. Control* **61**: 1327–1361.

Jung, S. & Wen, J. (2004). Nonlinear model predictive control for the swing-up of a rotary inverted pendulum, *ASME J. of Dynamic Systems, Measurement and Control* **126**(3): 666–673.

Kostin, G. V. & Saurin, V. V. (2006). The Optimization of the Motion of an Elastic Rod by the Method of Integro-Differential Relations, *Journal of computer and Systems Sciences International*, Vol. 45, Pleiades Publishing, Inc., pp. 217–225.

Lizarralde, F., Wen, J. & Hsu, L. (1999). A new model predictive control strategy for affine nonlinear control systems, *Proc of the American Control Conference (ACC '99), San Diego* pp. 4263 – 4267.

M. Bachmayer, J. R. & Ulbrich, H. (2008). Acceleration of linearly actuated elastic robots avoiding residual vibrations, *Proceedings of the 9th International Conference on Motion and Vibration Control, Munich, Germany*.

Magni, L. & Scattolini, R. (2004). Model predictive control of continuous-time nonlinear systems with piecewise constant control, *IEEE Transactions on automatic control* **49**(6): 900–906.

Schindele, D. & Aschemann, H. (2008). Nonlinear model predictive control of a high-speed linear axis driven by pneumatic muscles, *Proc. of the American Control Conference (ACC), 2008, Seattle, USA* pp. 3017–3022.

Shabana, A. A. (2005). *Dynamics of multibody systems*, Cambridge University Press, Cambridge.

Staudecker, M., Schlacher, K. & Hansl, R. (2008). Passivity based control and time optimal trajectory planning of a single mast stacker crane, *Proc. of the 17th IFAC World Congress, Seoul, Korea* pp. 875–880.

Wang, Y. & Boyd, S. (2010). Fast model predictive control using online optimization, *IEEE Transactions on control systems technology* **18**(2): 267–278.

Weidemann, D., Scherm, N. & Heimann, B. (2004). Discrete-time control by nonlinear online optimization on multiple shrinking horizons for underactuated manipulators, *Proceedings of the 15th CISM-IFToMM Symposium on Robot Design, Dynamics and Control, Montreal* .

Model Predictive Control

Edited by Tao Zheng

ISBN 978-953-307-102-2
Hard cover, 304 pages
Publisher Sciyo
Published online 18, August, 2010
Published in print edition August, 2010

Frontiers of Model Predictive Control Robust Model Predictive Control Nonlinear Model Predictive Control Excellent Applications Guide for Researchers and Engineers Recent Achievements of Authors over the World Theory with Practical Examples Kinds of Algorithms for Choice

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

IntechOpen