Multibody Syst Dyn (2012) 27:75–93
DOI 10.1007/s11044-011-9261-z

Two approaches for feedforward control and optimal design of underactuated multibody systems

Robert Seifried

Received: 4 November 2010 / Accepted: 3 May 2011 / Published online: 25 May 2011

Abstract An underactuated multibody system has less control inputs than degrees of freedom. For trajectory tracking, often a feedforward control is necessary. Two different approaches for feedforward control design are presented. The first approach is based on a coordinate transformation into the nonlinear input–output normal-form. The second approach uses servo-constraints and results in a set of differential algebraic equations. A comparison shows that both feedforward control designs have a similar structure. The analysis of the mechanical design of underactuated multibody systems might show that they are nonminimum phase, i.e., they have unstable internal dynamics. Then the feedforward control cannot be computed by time integration and output trajectory tracking becomes a very challenging task. Therefore, based on the two presented feedforward control design approaches, it is shown that through the use of an optimization procedure underactuated multibody systems can be designed in such a way that they are minimum phase. Thus, feedforward control design using the two approaches is significantly simplified.

Keywords Underactuation · Feedforward control · Model inversion · Servo-constraints · Internal dynamics · Minimum phase · Optimal design

1 Introduction

Multibody systems with less control inputs than degrees of freedom are called underactuated. Typical examples are multibody systems with passive joints, body flexibility, joint elasticities, and cranes. In order to obtain a good performance in end-effector trajectory tracking, an accurate and efficient feedforward control is necessary. The feedforward control design is based on an inverse model which provides the input required for exact reproduction of a desired output trajectory. In addition, the inverse model provides the trajectories of all generalized coordinates. In order to account for small disturbances and uncertainties, the feedforward control has to be supplemented by additional feedback control, whereby the

R. Seifried (✉)
Institute of Engineering and Computational Mechanics, University of Stuttgart, Pfaffenwaldring 9, 70569 Stuttgart, Germany
e-mail: robert.seifried@itm.uni-stuttgart.de

computed trajectories of the generalized coordinates can be used. This yields a so-called control structure with two design degrees of freedom, whereby both parts of this control system can be designed largely independent from each other.

Differentially flat systems [12] have the property that they can be inverted completely by algebraic computations without integration. Examples of differentially flat underactuated multibody systems are often cranes [12], cable manipulators [14], and manipulators with joint elasticities [10]. In contrast nonflat systems contain internal dynamics which must be solved. This increases the complexity of the feedforward control design. Examples are multibody systems with passive joints or body flexibility [20, 32]. Feedforward control design for those underactuated multibody systems with internal dynamics are considered in this paper.

The internal dynamics can be investigated using concepts from differential geometric control theory [16, 25, 34]. Thereby, a nonlinear coordinate transformation is used to transform the system symbolically into the so-called nonlinear input–output normal-form. This input–output normal-form has a favorable structure and a feedforward control can be derived easily, consisting of a chain of differentiators, driven internal dynamics, and an algebraic part. For systems with bounded internal dynamics, i.e., minimum phase systems, the internal dynamics can be solved by forward time integration [15]. In contrast, for nonminimum phase systems, the internal dynamics has to be computed off-line by solving a two-sided boundary value problem, yielding a noncausal control [11, 31, 36]. This approach of feedforward control design using a coordinate transformation is very efficient and also especially helpful for theoretical investigations. However, the complete symbolic derivation requires Lie-derivatives of the system output in state space representation, which is often only possible for small systems and special choices of the system output. In this paper, a simplified linearly-combined system output is chosen, which can be used to describe approximately the end-effector position of underactuated multibody systems with passive joints. Then it is shown that using this linearly combined output and the special structure of the equation of motion of a multibody system; the coordinate transformation can be directly performed on the second order differential equation of motion. In the practical implementation, this simplifies significantly the feedforward control design.

A second approach is presented which overcomes some of these shortcomings. This approach is based on servo-constraints to describe the trajectory tracking problem [2–5, 17, 21], which has been mostly applied to differentially flat mechanical systems. Similar to multibody systems with geometric constraints, this approach yields a set of differential-algebraic equations (DAE) which has to be solved numerically. However, in the case of servo-constraints, higher index DAE might occur [9]. Based on [28], it is shown in this paper that a servo-constraint approach for flat mechanical systems [4–7] can be extended to minimum phase systems with arbitrary output. In contrast to flat systems investigated in the mentioned literature, the nonflat systems which are treated in this paper require different numerical solution methods and a careful analysis and design of the internal dynamics. In the servo-constraint approach, a projection of the equation of motion into two orthogonal subspaces is performed. Then a comparison of the projected equations with the feedforward control approach using coordinate transformation shows that both feedforward control designs have the same structure. Also, it is seen that the DAE can only be solved by forward time integration if the system is minimum phase.

Due to these difficulties, the aim should be to design an underactuated multibody system in such a way that it is minimum phase [29, 30]. In this paper, two optimization-based design approaches for designing underactuated multibody systems with bounded internal dynamics are presented. These procedures are based on the two presented feedforward control design approaches. Thus, besides a minimum phase system design, also immediately

a viable feedforward control design is obtain from both approaches. In the proposed optimizations, the design parameters can be geometric dimensions and the mass distribution of the underactuated multibody systems. The optimization criteria has two steps and firstly requires that all eigenvalues of the linearized zero dynamics, this is the internal dynamics under constant zero output, are in the left half-plane. Secondly, it is required that initial errors in the nonlinear zero dynamics decay rapidly, i.e., the internal dynamics have good damping properties. Since this optimization problem is discontinuous, a particle-swarm optimization algorithm is used. The efficiency of both optimal design and feedforward control approaches is demonstrated for a multibody system with passive joints.

2 Feedforward control design using coordinate transformation

Underactuated multibody systems with f degrees of freedom, generalized coordinates $\boldsymbol{q} \in \mathbb{R}^f$ and inputs $\boldsymbol{u} \in \mathbb{R}^m$ with $m < f$, i.e., control forces and torques, are considered. The nonlinear equation of motion is given by

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{k}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \boldsymbol{g}(\boldsymbol{q}, \dot{\boldsymbol{q}}) + \boldsymbol{B}(\boldsymbol{q})\boldsymbol{u}, \tag{1}$$

where $\boldsymbol{M} \in \mathbb{R}^{f \times f}$ is the mass matrix, $\boldsymbol{k} \in \mathbb{R}^f$ the vector of generalized Coriolis, gyroscopic, and centrifugal forces and $\boldsymbol{g} \in \mathbb{R}^f$ the vector of applied forces. The control inputs $\boldsymbol{u}$ are projected on the directions of the generalized coordinates by the input matrix $\boldsymbol{B} \in \mathbb{R}^{f \times m}$. The system output $\boldsymbol{y} \in \mathbb{R}^m$ of the multibody system is given by

$$\boldsymbol{y} = \boldsymbol{h}(\boldsymbol{q}), \tag{2}$$

which, for example, can be an end-effector position. As often required in nonlinear control, the dimensions of input and output coincide. In the case of an underactuated multibody system, the input matrix $\boldsymbol{B}$ cannot be inverted. Therefore, the classical approach of inverse dynamics used in fully actuated systems [35] cannot be applied. Thus, more advanced nonlinear control techniques are necessary. In the following, a feedforward control design for output trajectory tracking of underactuated multibody systems is presented. This approach is based on concepts from differential geometry and its theoretical background for general nonlinear systems is described in [16, 25, 34].

2.1 Input–output normal-form

The nonlinear input–output normal-form is the basis for feedback linearization as well as for feedforward control design. This input–output normal-form is obtained by applying a nonlinear coordinate transformation to the equation of motion (1). This diffeomorphic coordinate transformation is given by $\boldsymbol{z} = \boldsymbol{\Phi}(\boldsymbol{x}) \in \mathbb{R}^{2f}$, where $\boldsymbol{x} = [\boldsymbol{q}^T, \dot{\boldsymbol{q}}^T]^T$ are the original states and $\boldsymbol{z} \in \mathbb{R}^{2f}$ are the states of the input–output normal-form. In general, this transformation requires a state-space representation and the symbolic computation of Lie-derivatives of the output $\boldsymbol{y}$ until the system input $\boldsymbol{u}$ occurs [16, 25, 34]. However, even for multibody systems with very few degrees of freedom, these symbolic calculations become very complicated. Therefore, here these calculations are directly performed on the second order equation of motion. The first two derivatives of the system output are

$$\dot{\boldsymbol{y}} = \boldsymbol{H}(\boldsymbol{q})\dot{\boldsymbol{q}}, \tag{3}$$

$$\ddot{\boldsymbol{y}} = \boldsymbol{H}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \bar{\boldsymbol{h}}(\boldsymbol{q}, \dot{\boldsymbol{q}}), \tag{4}$$

where $\boldsymbol{H}$ is the Jacobian-matrix of the system output and $\bar{\boldsymbol{h}} = \dot{\boldsymbol{H}}\dot{\boldsymbol{q}}$ is the local acceleration. In (4) the second derivative of the generalized coordinates $\ddot{\boldsymbol{q}}$ can be replaced by the equation of motion (1) and yields,

$$\ddot{\boldsymbol{y}} = \boldsymbol{H}\ddot{\boldsymbol{q}} + \bar{\boldsymbol{h}} = \boldsymbol{H}\boldsymbol{M}^{-1}[\boldsymbol{g} - \boldsymbol{k} + \boldsymbol{B}\boldsymbol{u}] + \bar{\boldsymbol{h}} = \boldsymbol{H}\boldsymbol{M}^{-1}\boldsymbol{B}\boldsymbol{u} + \boldsymbol{H}\boldsymbol{M}^{-1}[\boldsymbol{g} - \boldsymbol{k}] + \bar{\boldsymbol{h}}. \quad (5)$$

If the matrix $\boldsymbol{H}\boldsymbol{M}^{-1}\boldsymbol{B}$ is nonsingular, (5) can be solved for the control inputs $\boldsymbol{u}$. In this case, the matrix $\boldsymbol{H}\boldsymbol{M}^{-1}\boldsymbol{B}$ is called decoupling matrix and the system is said to have vector relative degree $r = \{r_1, \ldots, r_m\} = \{2, \ldots, 2\}$. Following [16], the relative degree is defined as the minimal number of derivatives of each system output $h_i(\boldsymbol{q}),\ i = 1, \ldots, m$ until the inputs $\boldsymbol{u}$ can be computed. Then no further derivatives are necessary and the first part of the coordinate transformation is found. Relative degree $r = \{2, \ldots, 2\}$ occurs in fully actuated multibody systems and is in many instances characteristic for multibody systems with passive joints or flexible bodies. In contrast, cranes and manipulators with joint elasticities have in most cases a higher relative degree. Then the nonlinear coordinate transformation is given by

$$z = \boldsymbol{\Phi}(\boldsymbol{x}) = \boldsymbol{\Phi}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix} \quad \begin{aligned} z_1 &= \boldsymbol{y} = \boldsymbol{h}(\boldsymbol{q}) \in \mathbb{R}^m, \\ \text{with } z_2 &= \dot{\boldsymbol{y}} = \boldsymbol{H}(\boldsymbol{q})\dot{\boldsymbol{q}} \in \mathbb{R}^m, \\ z_3 &= \boldsymbol{\Phi}_3(\boldsymbol{q}, \dot{\boldsymbol{q}}) \in \mathbb{R}^{2(f-m)}. \end{aligned} \quad (6)$$

Thereby, the coordinates z_3 are determined such that (6) forms at least a local diffeomorphic coordinate transformation. This requires that the Jacobian-matrix $\boldsymbol{J} = \partial\boldsymbol{\Phi}(\boldsymbol{x})/\partial\boldsymbol{x}$ is nonsingular. Applying the coordinate transformation (6) to the equation of motion (1) yields the nonlinear input–output normal-form

$$\boldsymbol{y} = z_1, \quad (7)$$

$$\dot{z}_1 = z_2, \quad (8)$$

$$\dot{z}_2 = \boldsymbol{H}\boldsymbol{M}^{-1}\boldsymbol{B}\boldsymbol{u} + \boldsymbol{H}\boldsymbol{M}^{-1}[\boldsymbol{g} - \boldsymbol{k}] + \bar{\boldsymbol{h}} = \boldsymbol{\beta}(z)\boldsymbol{u} + \boldsymbol{\alpha}(z), \quad (9)$$

$$\dot{z}_3 = \boldsymbol{\rho}(z) + \boldsymbol{\sigma}(z)\boldsymbol{u}. \quad (10)$$

The input–output normal-form consists of two parts. The first part, given by the output equation (7) and the differential equations (8)–(9), describe the relationship between the inputs $\boldsymbol{u}$ and outputs $\boldsymbol{y}$. Neglecting the output equation, this subsystem has dimension $2m$. The second part of the normal-form given by (10) has dimension $2(f - m)$ and describes the so-called internal dynamics. From this normal-form, the analysis of the internal dynamics, feedback linearization and feedforward control design can be performed.

It should be noted that for so-called differential flat systems [12, 23], such as many cranes or manipulators with joint elasticity, no internal dynamics remain. Such systems can be inverted completely by algebraic calculations. However, thereby higher derivatives of the system outputs $\boldsymbol{h}(\boldsymbol{q})$ have to be computed for the coordinate transformation, yielding a higher relative degree.

The analysis of the stability of the internal dynamics is crucial for control design. Since this analysis is often quite complex, the concept of zero dynamics is used in drawing important conclusions about the stability of the internal dynamics. The zero dynamics is the internal dynamics under the constraint that the output is kept constant, for example, identically zero such that $\boldsymbol{y} = z_1 = z_2 = \dot{z}_2 = \boldsymbol{0},\ \forall t$. For the considered underactuated multibody

systems, the required control input for this task follows from (9) as

$$\boldsymbol{u}_0 = \boldsymbol{\beta}^{-1}(\mathbf{0}, \mathbf{0}, z_3)\boldsymbol{\alpha}(\mathbf{0}, \mathbf{0}, z_3). \tag{11}$$

Applying this input $\boldsymbol{u}_0$ to (10), the internal dynamics reduces to the zero dynamics of the underactuated multibody system and reads

$$\dot{z}_3 = \boldsymbol{\rho}(\mathbf{0}, \mathbf{0}, z_3) + \boldsymbol{\sigma}(\mathbf{0}, \mathbf{0}, z_3)\boldsymbol{\beta}^{-1}(\mathbf{0}, \mathbf{0}, z_3)\boldsymbol{\alpha}(\mathbf{0}, \mathbf{0}, z_3). \tag{12}$$

A nonlinear system is called asymptotically minimum phase if the equilibrium point of the zero dynamics is asymptotically stable. Otherwise, the system is called nonminimum phase. For a first analysis, Lyapunov's indirect method can be used. If all eigenvalues of the linearized zero dynamics are in the open left half-plane, the zero dynamics is locally asymptotically stable, and thus the system is locally minimum phase [16, 25]. It should be noted, that the eigenvalues of the linearized zero dynamics coincide with the transmission zeros of the transfer function computed from the linearization of the original equation of motion (1); see [16]. The minimum phase property is invariant under a diffeomorphic coordinate transformation $\boldsymbol{z} = \boldsymbol{\Phi}(\boldsymbol{x})$. However, the minimum phase property depends on the system dynamics given by the equation of motion (1) and the choice of the system output (2).

2.2 Feedforward control design

An inverse model for feedforward control can be derived from the input–output normal-form (7)–(10). The desired trajectory must be at least twice differentiable yielding $z_1 = \boldsymbol{y}_d$, $z_2 = \dot{\boldsymbol{y}}_d$, $\dot{z}_2 = \ddot{\boldsymbol{y}}_d$. Then the required input $\boldsymbol{u}_d$ follows from (9) as

$$\boldsymbol{u}_d = \boldsymbol{\beta}^{-1}(\boldsymbol{y}_d, \dot{\boldsymbol{y}}_d, z_3)\big[\ddot{\boldsymbol{y}}_d - \boldsymbol{\alpha}(\boldsymbol{y}_d, \dot{\boldsymbol{y}}_d, z_3)\big]. \tag{13}$$

The computation of the input $\boldsymbol{u}_d$ depends on the desired output $\boldsymbol{y}_d$, $\dot{\boldsymbol{y}}_d$, $\ddot{\boldsymbol{y}}_d$ and the states of the internal dynamics z_3. These latter ones are the solution of the internal dynamics of (10) which is driven by $\boldsymbol{y}_d$, $\dot{\boldsymbol{y}}_d$ and $\boldsymbol{u}_d$. Replacing $\boldsymbol{u}_d$ in the internal dynamics (10) by (13) yields for the coordinates z_3 the differential equation

$$\dot{z}_3 = \boldsymbol{\rho}(\boldsymbol{y}_d, \dot{\boldsymbol{y}}_d, z_3) + \boldsymbol{\sigma}(\boldsymbol{y}_d, \dot{\boldsymbol{y}}_d, z_3)\boldsymbol{\beta}^{-1}(\boldsymbol{y}_d, \dot{\boldsymbol{y}}_d, z_3)\big[\ddot{\boldsymbol{y}}_d - \boldsymbol{\alpha}(\boldsymbol{y}_d, \dot{\boldsymbol{y}}_d, z_3)\big]. \tag{14}$$

In summary, the inverse model consists of three parts. The first part represents a chain of two differentiators for the desired output vector $\boldsymbol{y}_d$, producing the values $\dot{\boldsymbol{y}}_d$ and $\ddot{\boldsymbol{y}}_d$. The second part of the inverse model is the driven internal dynamics (14) for the z_3 coordinates. The third part of the inverse model is the algebraic equation (13) which computes from these values the desired input $\boldsymbol{u}_d$.

Several methods for model inversion exist which differ in the solution of the internal dynamics (14). In classical model inversion [15], the z_3 variables are found through forward integration of the internal dynamic (14) from the starting time point t_0 to the final time point t_f, using the initial values $z_3(t_0) = z_{3_0}$. However, in order to use the input $\boldsymbol{u}_d$ in a feedforward control, it must be bounded. Thus, depending on the stability of the internal dynamics, its forward integration might yield unbounded z_3 values, and thus unbounded inputs $\boldsymbol{u}_d$. Therefore, classical inversion can only be used for feedforward control design if the internal dynamics (14) remain bounded, which implies that only minimum phase systems

can be treated. In the case of nonminimum phase systems a bounded feedforward control can be computed by stable inversion as described in [11, 31]. However, in this approach, the internal dynamics (14) must be solved as a two-sided boundary value problem, e.g., using finite differences [36], and yields a noncausal solution.

2.3 Systems with linearly combined output

Even for multibody systems with very few degrees of freedom, the previously presented symbolic calculations can become very complicated. Especially for the derivation of the internal dynamics, it might be necessary that $\boldsymbol{z} = \boldsymbol{\Phi}(\boldsymbol{x})$ must be solved analytically for $\boldsymbol{x}$, which is difficult for general nonlinear functions $\boldsymbol{h}(\boldsymbol{q})$. Therefore, in the following, it is shown that for a special type of system output $\boldsymbol{y}$ the nonlinear input–output normal-form can be easily derived in symbolic form. In a first step, the equation of motion (1) is partitioned:

$$\begin{bmatrix} \boldsymbol{M}_{aa}(\boldsymbol{q}) & \boldsymbol{M}_{au}(\boldsymbol{q}) \\ \boldsymbol{M}_{au}^T(\boldsymbol{q}) & \boldsymbol{M}_{uu}(\boldsymbol{q}) \end{bmatrix} \begin{bmatrix} \ddot{\boldsymbol{q}}_a \\ \ddot{\boldsymbol{q}}_u \end{bmatrix} + \begin{bmatrix} \boldsymbol{k}_a(\boldsymbol{q}, \dot{\boldsymbol{q}}) \\ \boldsymbol{k}_u(\boldsymbol{q}, \dot{\boldsymbol{q}}) \end{bmatrix} = \begin{bmatrix} \boldsymbol{g}_a(\boldsymbol{q}, \dot{\boldsymbol{q}}) \\ \boldsymbol{g}_u(\boldsymbol{q}, \dot{\boldsymbol{q}}) \end{bmatrix} + \begin{bmatrix} \boldsymbol{B}_a(\boldsymbol{q}) \\ \boldsymbol{B}_u(\boldsymbol{q}) \end{bmatrix} \boldsymbol{u}. \tag{15}$$

Thereby, the submatrix $\boldsymbol{B}_a \in \mathbb{R}^{m \times m}$ has rank m. The first m rows of the partitioned equation of motion (15) are referred to as actuated part associated with the actuated coordinates $\boldsymbol{q}_a \in \mathbb{R}^m$. The remaining $f - m$ rows are referred to as the unactuated part associated with the unactuated coordinates $\boldsymbol{q}_u \in \mathbb{R}^{f-m}$. In the following, it is assumed that $\boldsymbol{B}_a = \boldsymbol{I}$ is the identity matrix and $\boldsymbol{B}_u = \boldsymbol{0}$. These special choices represent interesting cases of underactuated multibody systems in tree structure. Examples include rigid multibody systems with passive joints [31] and planar elastic manipulators, where the shape functions of the elastic bodies are chosen according to clamped boundary conditions; see, e.g., [20, 32].

The nonlinear input–output normal-form depends on the choice of the system output $\boldsymbol{y}$. Here, it is assumed that the end-effector position can be approximately described by an output of form

$$\boldsymbol{y} = \boldsymbol{h}(\boldsymbol{q}) = \boldsymbol{q}_a + \boldsymbol{\Gamma} \boldsymbol{q}_u, \tag{16}$$

where $\boldsymbol{\Gamma} \in \mathbb{R}^{m \times f_u}$. This output is a linear combination of actuated and unactuated generalized coordinates. For example, such an output can be used for elastic manipulators as shown in [20, 32] or systems with passive joints [31], which is also demonstrated in Sect. 5. The output (16) includes also the so-called collocated output $\boldsymbol{y} = \boldsymbol{q}_a$, i.e., $\boldsymbol{\Gamma} = \boldsymbol{0}$. From this, the coordinates of the input–output normal-form follow as:

$$\begin{aligned} z_1 &= \boldsymbol{y} = \boldsymbol{q}_a + \boldsymbol{\Gamma} \boldsymbol{q}_u, \\ z_2 &= \dot{\boldsymbol{y}} = \dot{\boldsymbol{q}}_a + \boldsymbol{\Gamma} \dot{\boldsymbol{q}}_u, \\ z_3 &= \left[\boldsymbol{q}_u^T, \dot{\boldsymbol{q}}_u^T \right]^T. \end{aligned} \tag{17}$$

Computing the Jacobian-matrix of the coordinate transformation shows that this choice of z_3 forms a diffeomorphic coordinate transformation. The partitioned equation of motion (15) is now transformed into the input–output normal-form; see, e.g., [31] for details. The input–output normal-form of the underactuated multibody system with the linearly combined system output reads then

$$\widetilde{\boldsymbol{M}} \ddot{\boldsymbol{y}} = \widetilde{\boldsymbol{g}} - \widetilde{\boldsymbol{k}} + \boldsymbol{u}, \tag{18}$$

$$\left[\boldsymbol{M}_{uu} - \boldsymbol{M}_{au}^T \boldsymbol{\Gamma} \right] \ddot{\boldsymbol{q}}_u = \boldsymbol{g}_u - \boldsymbol{k}_u - \boldsymbol{M}_{au}^T \widetilde{\boldsymbol{M}}^{-1} \left[\widetilde{\boldsymbol{g}} - \widetilde{\boldsymbol{k}} + \boldsymbol{u} \right]. \tag{19}$$

In this nonlinear input–output normal-form, the terms are summarized according to the convention

$$\begin{aligned}\widetilde{\boldsymbol{M}} &= \boldsymbol{M}_{aa} - (\boldsymbol{M}_{au} - \boldsymbol{M}_{aa}\boldsymbol{\Gamma})\big(\boldsymbol{M}_{uu} - \boldsymbol{M}_{au}^T\boldsymbol{\Gamma}\big)^{-1}\boldsymbol{M}_{au}^T,\\ \widetilde{\boldsymbol{g}} &= \boldsymbol{g}_a - (\boldsymbol{M}_{au} - \boldsymbol{M}_{aa}\boldsymbol{\Gamma})\big(\boldsymbol{M}_{uu} - \boldsymbol{M}_{au}^T\boldsymbol{\Gamma}\big)^{-1}\boldsymbol{g}_u,\\ \widetilde{\boldsymbol{k}} &= \boldsymbol{k}_a - (\boldsymbol{M}_{au} - \boldsymbol{M}_{aa}\boldsymbol{\Gamma})\big(\boldsymbol{M}_{uu} - \boldsymbol{M}_{au}^T\boldsymbol{\Gamma}\big)^{-1}\boldsymbol{k}_u.\end{aligned}$$

Equation (18) describes the relationship between the input $\boldsymbol{u}$ and output $\boldsymbol{y}$. The second part of the normal-form (19) describes the internal dynamics.

The control input for keeping the output identically zero, i.e. $\boldsymbol{y} = \boldsymbol{0}, \forall t$, follows from (18) as:

$$\boldsymbol{u}_0 = \widetilde{\boldsymbol{k}}(\boldsymbol{0}, \boldsymbol{q}_u, \boldsymbol{0}, \dot{\boldsymbol{q}}_u) - \widetilde{\boldsymbol{g}}(\boldsymbol{0}, \boldsymbol{q}_u, \boldsymbol{0}, \dot{\boldsymbol{q}}_u). \tag{20}$$

Applying this input $\boldsymbol{u}_0$ to the internal dynamics (19) reduces this to the zero dynamics of the underactuated multibody system

$$\big[\boldsymbol{M}_{uu}(\boldsymbol{0}, \boldsymbol{q}_u) - \boldsymbol{M}_{au}^T(\boldsymbol{0}, \boldsymbol{q}_u)\boldsymbol{\Gamma}\big]\ddot{\boldsymbol{q}}_u = \boldsymbol{g}_u(\boldsymbol{0}, \boldsymbol{q}_u, \boldsymbol{0}, \dot{\boldsymbol{q}}_u) - \boldsymbol{k}_u(\boldsymbol{0}, \boldsymbol{q}_u, \boldsymbol{0}, \dot{\boldsymbol{q}}_u). \tag{21}$$

As shown in Sect. 2.2 the feedforward control can be derived from the nonlinear input–output normal-form (18)–(19). The input $\boldsymbol{u}_d$ computed by the feedforward control follows in this case from (18) as

$$\boldsymbol{u}_d = \widetilde{\boldsymbol{M}}(\boldsymbol{y}_d, \boldsymbol{q}_u)\ddot{\boldsymbol{y}}_d - \widetilde{\boldsymbol{g}}(\boldsymbol{y}_d, \boldsymbol{q}_u, \dot{\boldsymbol{y}}_d, \dot{\boldsymbol{q}}_u) + \widetilde{\boldsymbol{k}}(\boldsymbol{y}_d, \boldsymbol{q}_u, \dot{\boldsymbol{y}}_d, \dot{\boldsymbol{q}}_u). \tag{22}$$

The computation of the input $\boldsymbol{u}_d$ depends on the desired output $\boldsymbol{y}_d, \dot{\boldsymbol{y}}_d$ and the unactuated states $\boldsymbol{q}_u, \dot{\boldsymbol{q}}_u$. These latter ones are the solution of the internal dynamics of (19) which is driven by $\boldsymbol{y}_d, \dot{\boldsymbol{y}}_d$ and $\boldsymbol{u}_d$. Replacing $\boldsymbol{u}_d$ in the internal dynamics of (19) by (22) yields for the unactuated states $\boldsymbol{q}_u, \dot{\boldsymbol{q}}_u$ the differential equation

$$\begin{aligned}&\big[\boldsymbol{M}_{uu}(\boldsymbol{y}_d, \boldsymbol{q}_u) - \boldsymbol{M}_{au}^T(\boldsymbol{y}_d, \boldsymbol{q}_u)\boldsymbol{\Gamma}\big]\ddot{\boldsymbol{q}}_u\\ &\quad= \boldsymbol{g}_u(\boldsymbol{y}_d, \boldsymbol{q}_u, \dot{\boldsymbol{y}}_d, \dot{\boldsymbol{q}}_u) - \boldsymbol{k}_u(\boldsymbol{y}_d, \boldsymbol{q}_u, \dot{\boldsymbol{y}}_d, \dot{\boldsymbol{q}}_u) - \boldsymbol{M}_{au}^T(\boldsymbol{y}_d, \boldsymbol{q}_u)\ddot{\boldsymbol{y}}_d.\end{aligned} \tag{23}$$

A graphical representation of the inverse model is shown schematically in Fig. 1. First, a chain of two differentiators computes for the desired output vector $\boldsymbol{y}_d$ the values $\dot{\boldsymbol{y}}_d$ and $\ddot{\boldsymbol{y}}_d$. Then the inverse model contains the driven internal dynamics (23) for the $\boldsymbol{q}_u, \dot{\boldsymbol{q}}_u$ states. Finally, the algebraic equation (22) computes from these values the desired input $\boldsymbol{u}_d$.

3 Feedforward control design using servo-constraints

An alternative approach for feedforward control design is the use of so-called servo-constraints [2–5, 17, 21]. These servo-constraints can be seen as an extension of geometric constraints representing joints. If in a multibody system additional geometric constraints arise, e.g., due to a kinematic loop, the equation of motion (1) is supplemented by constraint equations yielding the set of differential-algebraic equation (DAE),

$$\begin{aligned}\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{k}(\boldsymbol{q}, \dot{\boldsymbol{q}}) &= \boldsymbol{g}(\boldsymbol{q}, \dot{\boldsymbol{q}}) + \boldsymbol{C}^T(\boldsymbol{q})\boldsymbol{\lambda},\\ \boldsymbol{c}_g(\boldsymbol{q}) &= \boldsymbol{0}.\end{aligned} \tag{24}$$

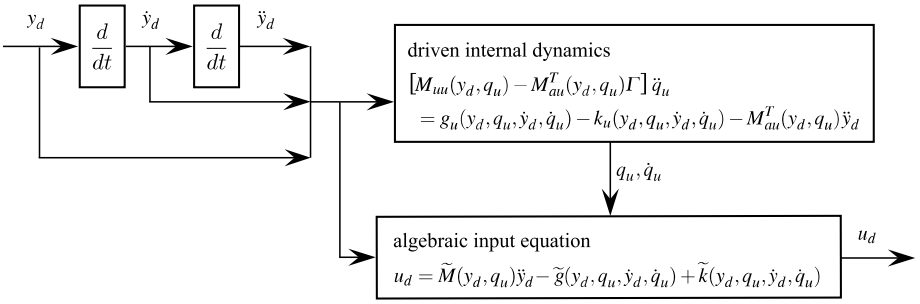

Fig. 1 Graphical representation of feedforward control of an underactuated MBS with linearly combined output

Thereby, $\boldsymbol{c}_g(\boldsymbol{q})$ is the constraint equation and $\boldsymbol{\lambda}$ the Lagrange multipliers associated with reaction forces due to the constraint. The matrix $\boldsymbol{C}=\partial\boldsymbol{c}_g(\boldsymbol{q})/\partial\boldsymbol{q}$ is the Jacobian-matrix of the constraint and $\boldsymbol{C}^T$ distributes the Lagrangian multipliers on the direction of the generalized coordinates.

In trajectory tracking, it is desired that the output $\boldsymbol{h}(\boldsymbol{q})$ tracks exactly the desired trajectories $\boldsymbol{y}_d$. This trajectory tracking problem can be described by supplementing the equation of motion (1) by the so-called servo-constraint $\boldsymbol{c}(\boldsymbol{q})$ such that

$$\begin{aligned}\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}}+\boldsymbol{k}(\boldsymbol{q},\dot{\boldsymbol{q}})&=\boldsymbol{g}(\boldsymbol{q},\dot{\boldsymbol{q}})+\boldsymbol{B}(\boldsymbol{q})\boldsymbol{u},\\ \boldsymbol{c}(\boldsymbol{q})=\boldsymbol{h}(\boldsymbol{q})-\boldsymbol{y}_d&=\boldsymbol{0}.\end{aligned}\tag{25}$$

Comparing (24) and (25) shows that both sets of DAEs have a very similar structure. The main difference is that in the case of servo-constraints the term $\boldsymbol{C}^T\boldsymbol{\lambda}$ is replaced by $\boldsymbol{B}\boldsymbol{u}$, where $\boldsymbol{u}$ are the unknown control inputs. Then the numerical solution of the DAE (25) provides the desired control inputs $\boldsymbol{u}_d$ and the trajectories of all generalized coordinates $\boldsymbol{q}$.

For the analysis of DAE problems, the differentiation index is often very helpful [8, 13]. Following [13], the differentiation index is the minimal number of analytical differentiations of the DAE (or parts of it) such that from the DAE an ordinary differential equation can be extracted. It is assumed that the geometric constraints $\boldsymbol{c}_g(\boldsymbol{q})$ are independent, i.e., the rows of $\boldsymbol{C}$ are linearly independent. Then after three differentiations of the constraint equation, a differential equation for $\boldsymbol{\lambda}$ can be extracted since the matrix $\boldsymbol{C}\boldsymbol{M}^{-1}\boldsymbol{C}^T$ has full rank; see, e.g., [13] for details. Thus, (24) has differentiation index 3.

In the case of servo-constraints, this is not any more necessarily true and higher index DAE can arise. Various mechanical systems with servo-constraints and different differentiation index are analyzed in [9]. As discussed there, the differentiation index is closely related to the previously discussed relative degree used in nonlinear control [16]. In [9], it is pointed out that the differentiation index is one higher than the relative degree if the internal dynamics are not affected by a constraint. In this paper, systems with vector relative degree $r=\{2,\ldots,2\}$ are treated. The corresponding DAE has therefore differentiation index 3, comparable to systems with geometric constraints (24). This is also seen from the nonsingular decoupling matrix $\boldsymbol{H}\boldsymbol{M}^{-1}\boldsymbol{B}$ following from (25), where $\boldsymbol{H}=\partial\boldsymbol{c}(\boldsymbol{q})/\partial\boldsymbol{q}$ is the Jacobian-matrix of the servo-constraints. The decoupling matrix corresponds to the matrix $\boldsymbol{C}\boldsymbol{M}^{-1}\boldsymbol{C}^T$ in the system with geometric constraints (24). Thus, the input matrix $\boldsymbol{B}$ projects the control inputs into directions with components orthogonal to the constraint manifold. Therefore, following [3–5] such a servo-constraint problem with nonsingular decoupling matrix $\boldsymbol{H}\boldsymbol{M}^{-1}\boldsymbol{B}$ is called an orthogonal realization.

For the considered systems, the DAE (25) might be solved directly with a DAE solver suitable for index 3 mechanical systems, e.g., [1, 19]. In [4, 5], feedforward control computation using servo-constraints for flat underactuated mechanical systems, such as cranes, is described. Thereby, higher index DAE arise and a projection method is used for index reduction. Then a backward Euler schema is used for the numerical solution of the index reduced system. For differentially flat flexible joint manipulators, a discrete time feedforward control is obtained in [17] by applying a backward Euler schema. For minimum phase manipulators with servo-constraints, [21] use higher order backward differentiation formula.

In this paper, the projection approach [4, 5] is extended to minimum phase underactuated multibody systems with vector relative degree $r = \{2, \ldots, 2\}$. By this, a one-to-one correspondence between the feedforward control using coordinate transformation and the feedforward control using servo-constraints is obtained. This correspondence is useful in the analysis of the feedforward control problem. In addition, index reduction is achieved by this approach. Differentiating the servo-constraint twice yields the constraint on velocity and acceleration level

$$\dot{\boldsymbol{c}} = \boldsymbol{H}(\boldsymbol{q})\dot{\boldsymbol{q}} - \dot{\boldsymbol{y}}_d = \boldsymbol{0}, \tag{26}$$

$$\ddot{\boldsymbol{c}} = \boldsymbol{H}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \bar{\boldsymbol{h}}(\boldsymbol{q}, \dot{\boldsymbol{q}}) - \ddot{\boldsymbol{y}}_d = \boldsymbol{0}. \tag{27}$$

As shown in [4], the equation of motion can be projected into two complementary subspaces in velocity space. These are the constrained and unconstrained subspace. The constrained subspace describes in this context the output subspace and follows from projection with the Jacobian-matrix of the output $\boldsymbol{H} \in \mathbb{R}^{m \times f}$. For the second subspace, an orthogonal complement $\boldsymbol{D} \in \mathbb{R}^{f \times f-m}$ must be derived, such that $\boldsymbol{H}\boldsymbol{D} = \boldsymbol{D}^T\boldsymbol{H}^T = \boldsymbol{0}$ is satisfied. Using these two matrices, the equation of motion (1) is projected into the two subspaces, yielding

$$\begin{bmatrix} \boldsymbol{H}\boldsymbol{M}^{-1} \\ \boldsymbol{D}^T \end{bmatrix} (\boldsymbol{M}\ddot{\boldsymbol{q}} + \boldsymbol{k} = \boldsymbol{g} + \boldsymbol{B}\boldsymbol{u}) \quad \Longrightarrow \quad \begin{aligned} &\boldsymbol{H}\ddot{\boldsymbol{q}} + \boldsymbol{H}\boldsymbol{M}^{-1}\boldsymbol{k} = \boldsymbol{H}\boldsymbol{M}^{-1}\boldsymbol{g} + \boldsymbol{H}\boldsymbol{M}^{-1}\boldsymbol{B}\boldsymbol{u}, \\ &\boldsymbol{D}^T\boldsymbol{M}\ddot{\boldsymbol{q}} + \boldsymbol{D}^T\boldsymbol{k} = \boldsymbol{D}^T\boldsymbol{g} + \boldsymbol{D}^T\boldsymbol{B}\boldsymbol{u}. \end{aligned} \tag{28}$$

From (27) follows the relationship $\boldsymbol{H}\ddot{\boldsymbol{q}} = \ddot{\boldsymbol{y}}_d - \bar{\boldsymbol{h}}$, which is applied to the first part of (28). Introducing the new state $\boldsymbol{v} = \dot{\boldsymbol{q}}$ in the projection results and adding the servo-constraints (25) on position level provides:

$$\boldsymbol{0} = \boldsymbol{H}\boldsymbol{M}^{-1}\boldsymbol{B}\boldsymbol{u} + \boldsymbol{H}\boldsymbol{M}^{-1}[\boldsymbol{g} - \boldsymbol{k}] + \bar{\boldsymbol{h}} - \ddot{\boldsymbol{y}}_d, \tag{29}$$

$$\dot{\boldsymbol{q}} = \boldsymbol{v}, \tag{30}$$

$$\boldsymbol{D}^T\boldsymbol{M}\dot{\boldsymbol{v}} = \boldsymbol{D}^T[\boldsymbol{g} - \boldsymbol{k}] + \boldsymbol{D}^T\boldsymbol{B}\boldsymbol{u}, \tag{31}$$

$$\boldsymbol{0} = \boldsymbol{h}(\boldsymbol{q}) - \boldsymbol{y}_d. \tag{32}$$

This forms a set of $2f + m$ differential algebraic equations for the $2f + m$ unknowns $\boldsymbol{q}, \boldsymbol{v}, \boldsymbol{u}$. Note that (29) has dimension m and describes an algebraic equation in $\boldsymbol{q}, \boldsymbol{v}, \boldsymbol{u}$. Together with the m servo-constraints (32) there are $2m$ algebraic equations in this DAE system. For the considered underactuated multibody systems, the decoupling matrix $\boldsymbol{H}\boldsymbol{M}^{-1}\boldsymbol{B}$ is nonsingular, and thus (29) can be solved for the desired input,

$$\boldsymbol{u}_d = \left(\boldsymbol{H}\boldsymbol{M}^{-1}\boldsymbol{B}\right)^{-1}\left[\ddot{\boldsymbol{y}}_d - \boldsymbol{H}\boldsymbol{M}^{-1}[\boldsymbol{g} - \boldsymbol{k}] - \bar{\boldsymbol{h}}\right]. \tag{33}$$

Inserting (33) in (31) results in

$$\dot{\boldsymbol{q}} = \boldsymbol{v}, \tag{34}$$

$$\boldsymbol{D}^T \boldsymbol{M} \dot{\boldsymbol{v}} = \boldsymbol{D}^T [\boldsymbol{g} - \boldsymbol{k}] + \boldsymbol{D}^T \boldsymbol{B} \left(\boldsymbol{H} \boldsymbol{M}^{-1} \boldsymbol{B} \right)^{-1} \left[\ddot{\boldsymbol{y}}_d - \boldsymbol{H} \boldsymbol{M}^{-1} [\boldsymbol{g} - \boldsymbol{k}] - \bar{\boldsymbol{h}} \right], \tag{35}$$

$$\boldsymbol{0} = \boldsymbol{h}(\boldsymbol{q}) - \boldsymbol{y}_d. \tag{36}$$

The numerical solution of this DAE provides the unknowns $\boldsymbol{q}, \boldsymbol{v}$, which are required to compute the control input $\boldsymbol{u}_d$ using (33).

In order to analyze the structure of the obtained feedforward control, a comparison with the input–output normal-form and the inverse model from Sect. 2 is done. Comparing (29) and (33) with (9) and (13), respectively, shows that they represent the same algebraic output equation. However, in the DAE approach this is given in original coordinates $\boldsymbol{q}, \boldsymbol{v}$, while (9) and (13) are given in the new coordinates $\boldsymbol{z} = [\boldsymbol{y}^T, \dot{\boldsymbol{y}}^T, \boldsymbol{z}_3^T]^T$. Comparing (34)–(36) with (14) shows that both describe the internal dynamics of the underactuated multibody system which is driven by the desired output trajectory $\boldsymbol{y}_d$ and its derivatives. Thereby, (14) is an ordinary differential equation of dimension $2(f - m)$ while (34)–(36) is a set of differential-algebraic equations, consisting of $2f - m$ differential equations and m servo-constraints. As discussed in Sect. 2, the internal dynamics can only be solved by forward time integration, if the system is minimum phase, otherwise unbounded states and inputs occur. Thus, in the case of minimum phase systems, the internal dynamics given by (34)–(36) can be solved by forward time integration. In summary, both approaches for feedforward control show the same structure, consisting of a chain of differentiators which computes the derivatives of the desired output trajectory $\boldsymbol{y}_d$, driven internal dynamics and an algebraic output equation. However, in the first approach presented in Sect. 2, the output tracking problem is solved by transforming the system into new coordinates containing the output $\boldsymbol{y}$, while in the second approach output tracking is achieved by introducing additional servo-constraints for the output $\boldsymbol{y}$.

Due to the projection and elimination of the control input $\boldsymbol{u}$ in (31) the obtained DAE (34)–(36) has index 2. By replacing the servo-constraints on position level (36) by the servo-constraints on velocity level (26), an index 1 DAE is obtained,

$$\begin{aligned} \dot{\boldsymbol{q}} &= \boldsymbol{v}, \\ \boldsymbol{D}^T \boldsymbol{M} \dot{\boldsymbol{v}} &= \boldsymbol{f}(\boldsymbol{q}, \boldsymbol{v}, t), \\ \boldsymbol{0} &= \boldsymbol{H} \boldsymbol{v} - \dot{\boldsymbol{y}}_d. \end{aligned} \tag{37}$$

Here, $\boldsymbol{f}(\boldsymbol{q}, \boldsymbol{v}, t)$ is the right-hand side of (35). This index 1 DAE can be solved easily, e.g., in Matlab using the *ode15s* integrator which is based on numerical differentiation formulas [33]. Since the servo-constraint on velocity level is used a drift of the servo-constraint can occur. However, for the example considered in this paper, the drift of the servo-constraints in the feedforward control design proofs to be negligible. Further, replacing the servo-constraints on position level (36) by the servo-constraints on acceleration level (27) yields an ordinary differential equation,

$$\begin{aligned} \dot{\boldsymbol{q}} &= \boldsymbol{v}, \\ \begin{bmatrix} \boldsymbol{D}^T \boldsymbol{M} \\ \boldsymbol{H} \end{bmatrix} \dot{\boldsymbol{v}} &= \begin{bmatrix} \boldsymbol{D}^T \\ \boldsymbol{H} \boldsymbol{M}^{-1} \end{bmatrix} \boldsymbol{M} \dot{\boldsymbol{v}} = \begin{bmatrix} \boldsymbol{f}(\boldsymbol{q}, \boldsymbol{v}, t) \\ \ddot{\boldsymbol{y}}_d - \bar{\boldsymbol{h}}(\boldsymbol{q}) \end{bmatrix}. \end{aligned} \tag{38}$$

It should be noted that the coefficient matrix coincides with the one used in (28), which is by construction nonsingular. Equation (38) can be solved with standard integrators for ordinary differential equations. Due to the use of the servo-constraints on acceleration level, a stronger drift might occur as in the index 1 DAE formulation.

For the numerical solution of the servo-constraint problem, the initial conditions must be consistent with the servo-constraints on position level (25) and velocity level (26). However, these conditions only provide $2m$ equations for the $2f$ states $\boldsymbol{q}, \boldsymbol{v}$. The remaining conditions are obtained here from the assumption, that the system starts at time t_0 from an equilibrium position, which is compatible with the servo-constraints. Thus, all states at time t_0 are specified.

For differential flat systems, such as presented in [4, 5], the matrix $\boldsymbol{H}\boldsymbol{M}^{-1}\boldsymbol{B}$ is singular, and thus (33) cannot be solved for $\boldsymbol{u}_d$, and thus the DAE (30)–(32) must be solved. Since flat systems can be inverted completely algebraically, the output specifies completely the entire motion of the system. Therefore, DAE (30)–(32) do not contain any internal dynamics. For the flat crane considered in [4], the initial DAE (25) has index 5 and the projected DAE (30)–(32) has index 3, which in this case can be solved efficiently by a backward Euler schema. However, such a backward Euler schema is not suitable for feedforward control design of underactuated multibody systems with internal dynamics. As shown in [28], the backward Euler schema introduces significant numerical damping into the internal dynamics and, therefore, provides inaccurate control inputs. This effect can only be moderated by using an extremely small time step size, which is computationally inefficient.

4 Optimization based design of underactuated multibody systems

As shown in Sects. 2 and 3 underactuated multibody systems might posses unbounded internal dynamics. Due to the shortcomings and difficulties in trajectory control of such non-minimum phase systems, it is desired to design the multibody system in such a way that the internal dynamics remains bounded. The internal dynamics depends on the choice of the system output $\boldsymbol{y}$ and the equation of motion (1) of the multibody system. Output relocation is a method where a different system output $\hat{\boldsymbol{y}}$ is chosen in order to achieve minimum phase property. However, the use of this approach is limited if trajectory tracking of an end-effector point is aspired. Thus, minimum phase property can only be achieved by modifying the system dynamics, which means the mechanical design of the underactuated multibody system must be altered. Such an optimization based design approach is proposed in [27, 29, 30]. This approach is originally based on the analysis and feedforward control design using coordinate transformation, as presented in Sect. 2 and summarized in the following. Then also the extension of this approach using the feedforward control by servo-constraints is presented.

4.1 Design procedure using coordinate transformation

As discussed in Sect. 2.1 the zero dynamics is often used to draw important conclusions about the internal dynamics. For example, the analysis of the zero dynamics (21) of underactuated manipulators with passive joints shows that the stability property depends on the mass distribution as well as on the geometric dimensions. For this analysis, the symbolic transformation into the input–output normal form is very helpful, since from these the dependencies are clearly seen. The mass distribution and geometry could be used directly as design variables $\boldsymbol{p}$. However, these quantities are mostly highly coupled, and the optimization might yield values which cannot be realized from an engineering point of view.

Therefore, other parametrizations of the mass distribution and geometry are more suitable; see examples in [29, 30] and in Sect. 5. A suitable parameterization of the optimization parameters are very problem specific and must be specified by the design engineer. However, the following proposed optimization procedure is general and independent of the chosen feasible parameterization of the design space.

The primary design goal is to achieve a stable zero dynamics, such that the underactuated multibody system is minimum phase. The zero dynamics given by (21) depends only on the unactuated states $\boldsymbol{q}_u, \dot{\boldsymbol{q}}_u$ and the design variables $\boldsymbol{p}$. Therefore, the zero dynamics can be written as

$$\left[\boldsymbol{M}_{uu}(\boldsymbol{p}, \boldsymbol{q}_u) - \boldsymbol{M}_{au}^T(\boldsymbol{p}, \boldsymbol{q}_u)\boldsymbol{\Gamma}\right]\ddot{\boldsymbol{q}}_u = \boldsymbol{g}_u(\boldsymbol{p}, \boldsymbol{q}_u, \dot{\boldsymbol{q}}_u) - \boldsymbol{k}_u(\boldsymbol{p}, \boldsymbol{q}_u, \dot{\boldsymbol{q}}_u). \tag{39}$$

In order to obtain a powerful mechanical design, not only minimum phase behavior must be guaranteed, but also additionally the zero dynamics should be designed in such a way that disturbances decay rapidly. This is especially important in order to avoid that disturbances yield large undesired vibrations of the internal dynamics during trajectory tracking. A two-step computation of the optimization criteria $f(\boldsymbol{p})$ is proposed, which should be minimized in the course of the optimization:

Step 1: In the first step, Lyapunov's indirect method is used for analyzing the asymptotic stability of the zero dynamics. Thus, all eigenvalues of the linearized zero dynamics must be in the left half-plane:

$$\mathrm{Re}\left[\lambda\left(\boldsymbol{A}(\boldsymbol{p})\right)\right] < 0, \tag{40}$$

where $\boldsymbol{A}(\boldsymbol{p})$ is the system matrix of the linearized zero dynamics (39). If at least one eigenvalue has a nonnegative real part, a large default value for the optimization criteria $f(\boldsymbol{p})$ is returned. Otherwise, the linearized analysis shows local asymptotic stability and it is proceeded with step 2.

Step 2: If all eigenvalues are in the left half-plane, the final optimization criteria $f(\boldsymbol{p})$ is calculated. In order to achieve good damping properties, it is required that initial errors in the nonlinear zero dynamics (39) decay rapidly. The disturbance is given by the initial conditions $\boldsymbol{q}_u(t_0) = \boldsymbol{q}_{u_0}, \dot{\boldsymbol{q}}_u(t_0) = \boldsymbol{0}$. The optimization criteria $f(\boldsymbol{p})$ is then described by the maximal cumulated squared error of the $f - m$ unactuated coordinates $\boldsymbol{q}_u$ with respect to the equilibrium point $\boldsymbol{q}_u = \boldsymbol{0}$ of the zero dynamics. This is given by

$$f(\boldsymbol{p}) = \max_i \int_{t_0}^{t_1} q_{u_i}^2 \, dt, \tag{41}$$

where t_1 describes the final time of the simulation. By evaluating only the attenuation of the least damped coordinate of the zero dynamics, it is achieved that the design improvements concentrates on the damping properties of the least damped coordinate. Besides evaluating the damping properties of the zero dynamics, this simulation also provides a very good indication about the behavior of the nonlinear zero dynamics. It gives an indication if the zero dynamics remains stable in the case that the internal states are pushed by a disturbance further away from the equilibrium point.

It should be noted that from a practical point of view this design approach yields a viable design. However, to prove that the states of the internal dynamics and the inputs remain bounded during trajectory tracking, a complete stability analysis of the driven internal dynamics for each specific trajectory would be necessary [16]. In this paper, this is verified implicitly by the computation of the feedforward control, which would fail if the states become unbounded.

4.2 Design procedure using servo-constraints

Instead of using the explicit transformation of the equation of motion into the input–output normal-form, also the feedforward control design approach using servo-constraints can be used in this design procedure. In this case, the same quantities are considered in the optimization criteria computation. Since they are computed in a different way, the differences are briefly summarized.

Step 1: The eigenvalues of the zero dynamics coincide with the zeros of the transfer function of the linearized system [16]. Thus, the complete equation of motion (1) is linearized and the zeros are computed. For details on system zeros, see, e.g., [37]. If all zeros are in the left half-plane, it is proceeded with step 2, otherwise a large default value is returned.

Step 2: The damping properties of the zero dynamics is evaluated using DAE (37). Therefore, the output trajectory $\boldsymbol{y}_d$ is kept constant for all time. Again a disturbance is introduced in the initial conditions of the unactuated coordinates $\boldsymbol{q}_u(t_0) = \boldsymbol{q}_{u_0}, \dot{\boldsymbol{q}}_u(t_0) = \boldsymbol{0}$ and the response is simulated. In order to obtain consistent initial conditions for the time integration, the actuated coordinates $\boldsymbol{q}_a(t_0) = \boldsymbol{q}_{a_0}, \dot{\boldsymbol{q}}_a(t_0) = \boldsymbol{0}$ must fulfill $\boldsymbol{h}(\boldsymbol{q}_{a_0}, \boldsymbol{q}_{u_0}) = \boldsymbol{y}_{d_0}$. Then the optimization criteria is again evaluated using (41).

4.3 Particle swarm optimization

In the optimization procedure, the criteria function $f(\boldsymbol{p})$ should be minimized with respect to the design variables $\boldsymbol{p}$. Due to the two-step criteria computation, the optimization problem is discontinuous and thus the gradient free particle swarm optimization procedure is used. This population based optimization method originates in the study and simulation of social behavior of bird and fish flocks; see [18]. The basic idea is the modeling of social interaction between individual particles of a population on the quest for the best point in the feasible design space. Thereby, it is aspired to use the collective intelligence of a swarm to solve complex optimization problems. The swarm is described by the parameter sets of all particles, where $\boldsymbol{p}_i$ is the set of the ith particle of the swarm. The recursive update equation is the basic step of the particle swarm optimization and reads for iteration step k,

$$\boldsymbol{p}_i^{k+1} = \boldsymbol{p}_i^k + \Delta \boldsymbol{p}_i^{k+1}, \tag{42}$$

$$\Delta \boldsymbol{p}_i^{k+1} = w \Delta \boldsymbol{p}_i^k + c_1 r_1 \left(\boldsymbol{p}_i^{\text{best},k} - \boldsymbol{p}_i^k\right) + c_2 r_1 \left(\boldsymbol{p}_{\text{swarm}}^{\text{best},k} - \boldsymbol{p}_i^k\right). \tag{43}$$

Thereby, $r_1,\ r_2$ are evenly distributed numbers and $w,\ c_1,\ c_2$ are control parameters. The update of the parameters $\Delta \boldsymbol{p}_i^{k+1}$ consists of three contributions. The first part describes the tradition and the particle moves in the direction of the previous update $\Delta \boldsymbol{p}_i^k$. The cognitive part moves the particle in the direction of the best parameter set $\boldsymbol{p}_i^{\text{best},k}$ which it found on its own. The social behavior part moves the particle in the direction of the best parameter set $\boldsymbol{p}_{\text{swarm}}^{\text{best},k}$ which the entire swarm has found so far.

Advantages of the particle swarm optimization are that no gradient information is necessary; the solution is independent of initial sets of design parameters $\boldsymbol{p}$, and there are no requirements on smoothness or continuity of the optimization criteria. This approach is well suited for finding global minima and is often easy to program and to adjust to specific problems. The used algorithm is a Matlab implementation presented in [26]. Compared to gradient based methods, a general disadvantage of stochastic optimization algorithms is their large computational expense due to a large amount of criteria evaluations. In the criteria computation, the most time consuming part is the time integration of the zero dynamics in

the second step. However, in the first step of the criteria computation, many unfeasible designs are filtered out and thus the number of time integrations is reduced by the restriction to locally stable designs.

5 Design and feedforward control of a manipulator with passive joints

The optimization-based design procedures and the feedforward control design approaches for underactuated multibody systems are demonstrated using an underactuated manipulator with two passive joints. Passive joints might result intentionally from a cost and efficiency driven reduction of the number of actuators, as used by [24] in the design of a hyper-articulated assembly arm. Also, such an approach has been proposed in the design of mechanical grippers or robotic hands, where in addition the underactuation supports the adaption to the shape of the object; see, e.g., [22]. The manipulator investigated in this paper is shown in Fig. 2. It consist of a chain of four links. The links 1 and 2 are actuated by torques $\boldsymbol{u} = [T_1, T_2]^T$. The links 3, 4 are passive and are supported by spring-damper combinations. At the end of link 4, a loading mass is added. The system is described by relative coordinates. The actuated generalized coordinates are $\boldsymbol{q}_a = [\alpha_1, \alpha_2]^T$ and the unactuated generalized coordinates are $\boldsymbol{q}_u = [\beta_1, \beta_2]^T$. The manipulator is made of aluminum and the physical properties of the initial design are summarized in Table 1.

The end-effector point of the manipulator should follow a predefined trajectory. In order to derive the input–output normal-form, the position of the end-effector point is described approximately by a linearly combined system output of form (16). For a somewhat stiff spring-damper combination, the angles β_1, β_2 remain small. Then the end-effector position

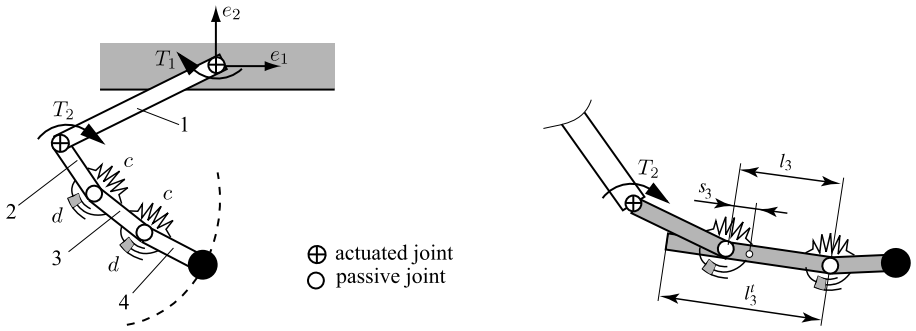

Fig. 2 Initial design of underactuated manipulator with two passive joints (*left*) and proposed design parametrization for the unactuated links (*right*)

Table 1 Initial parameters for underactuated manipulator

Link 1	$m_1 = 6.875$ kg	$I_1 = 0.5743\,\text{kg}\,\text{m}^2$	$l_1 = 1.0$ m
Link 2–4	$m_i = 2.292$ kg	$I_i = 0.0217\,\text{kg}\,\text{m}^2$	$l_i = 0.333$ m with $i = 2, 3, 4$
Load	$m_l = 6$ kg	$I_l = 0.0147\,\text{kg}\,\text{m}^2$	
Spring/damper	$c = 400$ Nm/rad	$d = 0.25$ Nms/rad	

can be approximated by using the linearly combined output $\boldsymbol{y}$ such that

$$\boldsymbol{r}^{\mathrm{ef}}(\boldsymbol{q}_a, \boldsymbol{q}_u) \approx \hat{\boldsymbol{r}}^{\mathrm{ef}}(\boldsymbol{y}) = \begin{bmatrix} l_1 \sin(y_1) + (l_2 + l_3 + l_4) \sin(y_1 + y_2) \\ -l_1 \cos(y_1) - (l_2 + l_3 + l_4) \cos(y_1 + y_2) \end{bmatrix}, \tag{44}$$

where $y_1 = \alpha_1$ and $y_2 = \alpha_2 + \Gamma_1\beta_1 + \Gamma_2\beta_2$. For small angles $\beta_1,\ \beta_2$, the system output y_2 can be seen as an auxiliary angle for describing the end-effector point. For the determination of Γ_1, Γ_2, the exact output $\boldsymbol{r}^{\mathrm{ef}}(\boldsymbol{q}_a, \boldsymbol{q}_u)$ and the approximated output $\hat{\boldsymbol{r}}^{\mathrm{ef}}(\boldsymbol{y})$ are linearized around a desired trajectory with $\beta_1 = \beta_2 = 0$. Then, from a comparison it follows that for $\Gamma_1 = (l_3 + l_4)/(l_2 + l_3 + l_4)$ and $\Gamma_2 = l_4/(l_2 + l_3 + l_4)$ the linearized outputs coincide. Using an inverse kinematics procedure, the desired output trajectories $\boldsymbol{y}_d(t)$ can be computed from the desired end-effector trajectory $\boldsymbol{r}_d^{\mathrm{ef}}(t)$ based on (44). However, it should be noted that due to this approximation a small tracking error for the end-effector position has to be expected. With the derived linearly combined system output, the input–output normal-form can be derived conveniently following the steps presented in Sect. 2.3.

An analysis of the zero dynamics shows that the initial design of the system is nonminimum phase. Also, the analysis shows that for this manipulator the linearized zero dynamics is independent of the position of the manipulator. Due to the nonminimum phase property, the proposed optimization procedure is used to design a minimum phase manipulator. The stability of the zero dynamics (39) of the underactuated manipulator depends on the mass distribution of the unactuated links $i = 3, 4$, given by their mass m_i, inertia I_{iz} and center of mass s_i, as well as by their geometry described by the length l_i. In the absence of gravity, the spring and damping coefficients c, d alter the response of the system, but not the stability. There are many feasible design parameterizations, which depend strongly on the given problem. In [29], a basic homogeneous initial design of the underactuated multibody system with constant link length is assumed and additional small balancing masses are added to each unactuated link in order to alter the stability property of the zero dynamics. With this approach, a minimum phase behavior is achieved using only a modest total mass increase. Thereby, the optimization yields designs where the additional masses are mounted as counterweights.

In this paper, a parameterization with focus on changing the length of the unactuated homogenous links l_3, l_4 is proposed. The length of the second link l_2 is chosen such that the total length of all three links is 1 m, i.e., $l_2 = 1m - l_3 - l_4$. However, it turns out that by only changing the link length no viable minimum phase design is found. Therefore, in addition to the change of link length also the center of gravity s_3 of the third link is introduced as a design variable. This is motivated by the results in [29] where the center of gravity of the unactuated links is moved closer to the passive joints by counterweights. This change of the center of mass can be achieved by extending link 3 in the opposite direction; see Fig. 2. Thus, the total length of the third link is $l_3' = 2(l_3 - s_3)$. Such an approach could also be used for link 4, however, it turns out that this additional design parameter does not influence the optimal design of this example. A design is aspired, where the links 2, 3, 4 are homogenous with the same constant cross section. In order to avoid too small cross sections, and to allow a detailed comparison with the initial design, the mass of the three links should remain constant compared to the initial design. Then the cross section is determined from this mass constraint. In order to obtain a viable physical design, bounds have to be put on the design variables which results in the feasible design space

$$P = \left\{ \boldsymbol{p} = \begin{bmatrix} l_3 \\ s_3 \\ l_4 \end{bmatrix} \middle| \, 0.3\ \mathrm{m} \le l_3 \le 0.7\ \mathrm{m},\ 0.1 \le s_3/l_3 \le 0.5,\ 0.1\ \mathrm{m} \le l_4 \le 0.3\ \mathrm{m} \right\}. \tag{45}$$

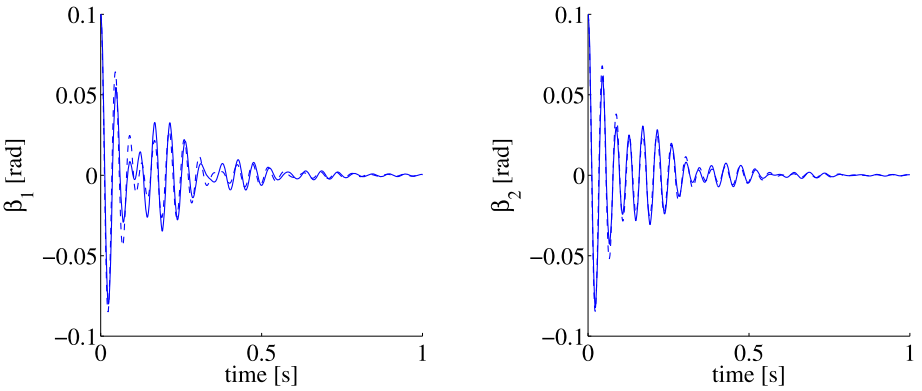

Fig. 3 Optimized internal dynamics under disturbance (*solid*: with coordinate transformation; *dashed*: servo-constraints)

Both design approaches yield the optimal parameters $l_3 = 0.334$ m, $s_3 = 0.034$, and $l_4 = 0.1$ m. Thus, the third link has total length $l_3^t = 0.6$ m. It is interesting to note, that the design variables are very close to their bounds. A detailed investigation shows that an extension of the length of link 4 yields nearly immediately a nonminimum phase design. It should be mentioned, that the obtained design is robust again some parameter uncertainties. For example, the obtained design remains its minimum phase property when varying the loading mass and inertia in the range of $0.1m_l - 5m_l$ and $0.1I_l - 5I_l$, respectively. For the optimized parameter set, the damping behavior of the zero dynamics under a disturbance is shown in Fig. 3. This shows that a good attenuation of the initial disturbance is achieved by the optimized design. Also, this simulated dynamic response shows for both approaches an overall good agreement. There are some smaller differences, which result from the use of a linearly combined output in the coordinate transformation approach. When using this linearly combined system output also in the servo-constraint approach, identical results are obtained with both feedforward control design approaches. However, this simulation of the response of the optimized design under disturbance is about 3.5 times faster using the coordinate transformation approach than the servo-constraint approach with index 1 formulation.

For the obtained optimal design, the feedforward control is computed and tested by simulation. Using the coordinate transformation approach, the feedforward control is computed using (22) and (23). For the servo-constraint approach the index 1 DAE formulation (37) is used. In the example, the end-effector should follow a half-circular end-effector trajectory. The center of the half-circle is at position $(0, -1.5\text{ m})$ and the radius is 0.75 m; see Fig. 4. The end-effector point should follow the trajectory in the short time period of 1.5 s, which describes an aggressive manoeuver.

Figure 5 shows the control inputs computed with both feedforward control approaches and the obtained end-effector error. Both feedforward control approaches yield nearly identical results, with no visible differences in the control input. Due to the approximation of the end-effector point by a linearly combined output in the coordinate transformation approach, a tracking error of 0.4 mm occurs. In contrast, using the servo-constraint approach the end-effector point is used directly. Therefore, a smaller tracking error of 0.002 mm is achieved, which is mainly due to numerical errors. For this trajectory, the computation time is again approximately 3.5 times faster using the coordinate transformation approach compared to the servo-constraint approach. The vibrations seen in the control inputs at the beginning and

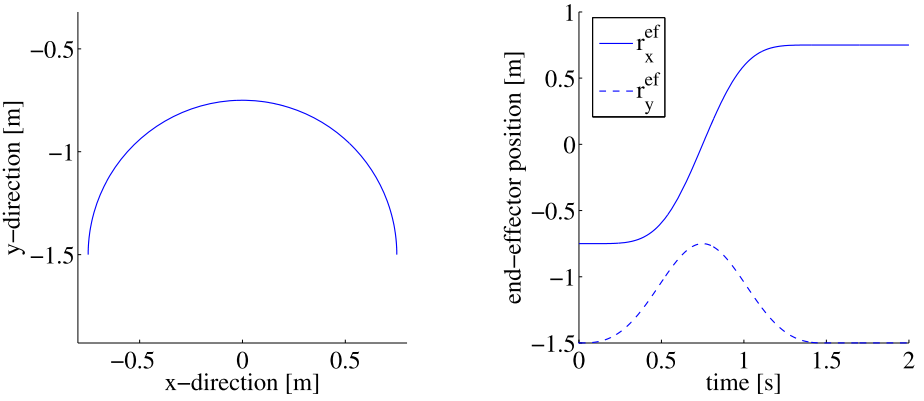

Fig. 4 Desired end-effector trajectory

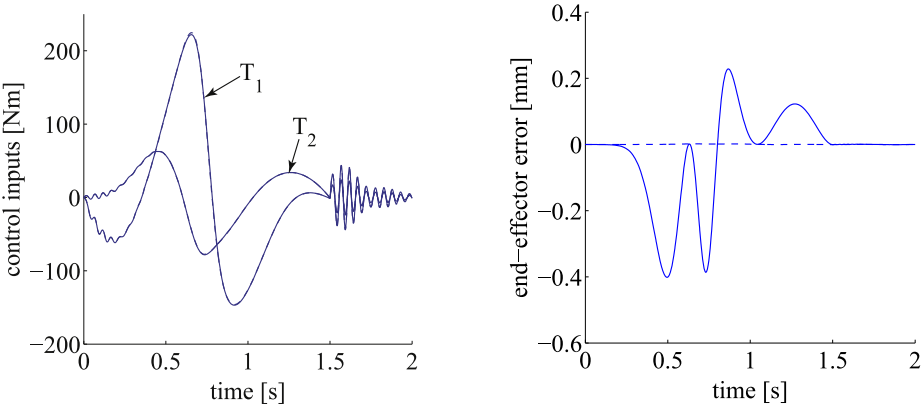

Fig. 5 Computed control inputs and end-effector trajectory error (*solid*: with coordinate transformation; *dashed*: servo-constraints)

after ending of the trajectory tracking at 1.5 s result from the internal dynamics. It shows again that the optimization yields a design whose internal dynamics have good damping properties. It should be noted, that with the obtained design the feedforward control can also be computed for the case of no physical damping in the passive joint, i.e., $d = 0$. However, than the internal dynamics is nearly undamped and yields strong oscillations in the unactuated coordinates and control inputs.

6 Conclusions

Two approaches for feedforward control design of minimum phase underactuated multibody systems are presented. The first method is based on the nonlinear coordinate transformation of the equation of motion into the nonlinear input–output normal-form from which the inverse model is deducted. The second method is based on the description of the desired output trajectory tracking problem using servo-constraints, resulting in a DAE system.

In both cases, the inverse model consists of a chain of differentiators, driven internal dynamics and an algebraic equation. For minimum phase systems, forward time integration can be used in both approaches to compute the feedforward control. The first feedforward control design approach requires larger symbolic manipulations for the coordinate transformation, for which an approximation of the desired output might be necessary. In contrast, the servo-constraint approach can be easily applied to arbitrary outputs and requires less precomputations. However, due to the required solution of a DAE, this approach is numerically less efficient and might be numerically more sensitive. Both feedforward control approaches can be used for the optimization based design of minimum phase underactuated multibody systems. These design approaches are based on the optimization of the zero dynamics, and yields stable internal dynamics with good damping properties. Design parameters are the mass distributions or geometric dimensions of the unactuated bodies. Due to the discontinuous optimization criteria, a stochastic particle swarm optimization algorithm is used. Then, for trajectory tracking of the optimal designs, the two presented feedforward control design approaches can be used. The two approaches agree very well and their accuracy is demonstrated by simulation using a planar underactuated manipulator with two passive joints.

Acknowledgements The author would like to thank the German Research Foundation (DFG) for financial support of the project within the Cluster of Excellence in Simulation Technology (EXC 310/1) at the University of Stuttgart.

References

1. Arnold, M., Brüls, O.: Convergence of the generalized-α scheme for constrained mechanical systems. Multibody Syst. Dyn. **18**, 185–202 (2007)
2. Betsch, P., Uhlar, S., Quasem, M.: On the incorporation of servo constraints into a rotationless formulation of flexible multibody dynamics. In: Proceedings of the ECCOMAS Thematic Conference Multibody Dynamics, Milano, Italy, 25–28 June 2007
3. Blajer, W.: Index of differential-algebraic equations governing the dynamics of constraint mechanical systems. Appl. Math. Model. **16**, 70–77 (1992)
4. Blajer, W., Kolodziejczyk, K.: A geometric approach to solving problems of control constraints: theory and a DAE framework. Multibody Syst. Dyn. **11**, 343–364 (2004)
5. Blajer, W., Kolodziejczyk, K.: Control of underactuated mechanical systems with servo-constraints. Nonlinear Dyn. **50**, 781–791 (2007)
6. Blajer, W., Kolodziejczyk, K.: Improved DAE formulation for inverse dynamics simulation of cranes. Multibody Syst. Dyn. **25**, 131–143 (2011)
7. Blajer, W., Dziewiecki, K., Kolodziejczyk, K., Mazur, Z.: Inverse dynamics of underactuated mechanical systems: a simple case study and experimental verification. Commun. Nonlinear Sci. Numer. Simul. **16**, 2265–2272 (2011)
8. Brenan, K., Campbell, S., Petzold, L.: Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations. SIAM, Philadelphia (1996)
9. Campbell, S.: High-index differential algebraic equations. Mech. Based Des. Struct. Mach. **23**, 199–222 (1995)
10. De Luca, A., Book, W.: Robots with flexible elements. In: Handbook of Robotics, pp. 287–319. Springer, Berlin (2008), Chap. 13
11. Devasia, S., Chen, D., Paden, B.: Nonlinear inversion-based output tracking. IEEE Trans. Autom. Control **41**, 930–942 (1996)
12. Fliess, M., Lévine, J., Martin, P., Rouchon, P.: Flatness and defect of nonlinear systems: introductory theory and examples. Int. J. Control **61**, 1327–1361 (1995)
13. Hairer, E., Wanner, G.: Solving Ordinary Differential Equations II—Stiff and Differential Algebraic Problems. Springer, Berlin (2010)
14. Heyden, T., Woernle, C.: Dynamics and flatness-based control of a kinematically undetermined cable suspension manipulator. Multibody Syst. Dyn. **16**, 155–177 (2006)
15. Hirschorn, R.: Invertibility of multivariable nonlinear control systems. IEEE Trans. Autom. Control **24**, 855–865 (1979)

16. Isidori, A.: Nonlinear Control Systems. Springer, London (1995)
17. Jankowski, K., Van Brussel, H.: An approach to discrete inverse dynamics control of flexible-joint robots. IEEE Trans. Robot. Autom. **8**, 651–658 (1992)
18. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: Proceedings of the International Conference on Neural Networks, Perth, Australia, pp. 1942–1948 (1995)
19. Leyendecker, S., Marsen, J., Oritz, M.: Variational integrators for constraint dynamical systems. Z. Angew. Math. Mech. **88**, 677–708 (2008)
20. Moallem, M., Patel, R., Khorasani, K.: Flexible-Link Robot Manipulators: Control Techniques and Structural Design. Springer, London (2000)
21. Moberg, S., Hanssen, S.: A DAE approach to feedforward control of flexible manipulators. In: Proceedings of the 2007 IEEE International Conference on Robotics and Automation, pp. 3439–3444 (2007)
22. Montambault, S., Gosselin, C.: Analysis of underactuated mechanical grippers. J. Mech. Des. **123**, 367–374 (2001)
23. Rouchon, P.: Flatness based control of oscillators. J. Appl. Math. Mech. **85**, 411–421 (2005)
24. Roy, B., Asada, H.: Nonlinear feedback control of a gravity-assisted underactuated manipulator with application to aircraft assembly. IEEE Trans. Robot. **25**, 1125–1133 (2009)
25. Sastry, S.: Nonlinear Systems: Analysis, Stability and Control. Springer, New York (1999)
26. Sedlaczek, K., Eberhard, P.: Using augmented Lagrangian particle swarm optimization for constrained problems in engineering. Struct. Multidiscip. Optim. **32**, 277–286 (2006)
27. Seifried, R.: Optimization-based design of feedback linearizable underactuated multibody systems. In: Proceedings of the ECCOMAS Thematic Conference Multibody Dynamics 2009, Warsaw, Poland, 29 June–2 July (2009), paper ID 121
28. Seifried, R.: Two approaches for designing minimum phase underactuated multibody systems. In: Proceedings of the First Joint International Conference on Multibody System Dynamics, Lappeenranta, Finland (2010)
29. Seifried, R.: Optimization-based design of minimum phase underactuated multibody systems. In: Arczewski, K., Blajer, W., Fraczek, J., Wojtyra, M. (eds.) Multibody Dynamics: Computational Methods and Applications, pp. 261–282. Springer, Berlin (2011)
30. Seifried, R.: Integrated mechanical and control design of underactuated multibody systems. Nonlinear Dyn. (accepted for publication). doi:10.1007/s11071-011-0087-2
31. Seifried, R., Eberhard, P.: Design of feed-forward control for underactuated multibody systems with kinematic redundancy. In: Ulbrich, H., Ginzinger, L. (eds.) Motion and Vibration Control: Selected Papers from MOVIC 2008, pp. 275–284. Springer, Berlin (2009)
32. Seifried, R., Held, A., Dietmann, F.: Analysis of feed-forward control design approaches for flexible multibody systems. J. Syst. Des. Dyn. **5**, 429–440 (2011)
33. Shampine, L., Reichelt, M., Kierzenka, J.: Solving index-1 DAEs in MATLAB and Simulink. SIAM Rev. **18**, 538–552 (1999)
34. Soltine, J.-J., Li, W.: Applied Nonlinear Control. Prentice-Hall, Englewood Cliffs (1991)
35. Spong, M., Hutchinson, S., Vidyasagar, M.: Robot Modeling and Control. Wiley, New York (2006)
36. Taylor, D., Li, S.: Stable inversion of continuous-time nonlinear systems by finite-difference methods. IEEE Trans. Autom. Control **47**, 537–542 (2002)
37. Zhou, K., Doyle, J., Glover, K.: Robust and Optimal Control. Prentice-Hall, Upper Saddle River (1996)