

# Inverse Dynamics of Serial and Parallel Underactuated Multibody Systems using a DAE Optimal Control Approach

Guaraci Jr. Bastos      Robert Seifried      Olivier Brüls

April 18, 2013

## Abstract

The inverse dynamics analysis of underactuated multibody systems aims at determining the control inputs in order to track a prescribed trajectory. This paper studies the inverse dynamics of non-minimum phase underactuated multibody systems with serial and parallel planar topology, e.g. for end-effector control of flexible manipulators or manipulators with passive joints. Unlike for minimum phase systems, the inverse dynamics of non-minimum phase systems cannot be solved by adding trajectory constraints (servo-constraints) to the equations of motion and applying a forward time integration. Indeed, the inverse dynamics of a non-minimum phase system is known to be non-causal, which means that the control forces and torques should start before the beginning of the trajectory (pre-actuation phase) and continue after the end-point is reached (post-actuation phase). The existing stable inversion method proposed for general nonlinear non-minimum phase systems requires to derive explicitly the equations of the internal dynamics and to solve a boundary value problem. This paper proposes an alternative solution strategy which is based on an optimal control approach using a direct transcription method. The method is illustrated for the inverse dynamics of an underactuated serial manipulator with rigid links and four degrees-of-freedom and an underactuated parallel machine. An important advantage of the proposed approach is that it can be applied directly to the standard equations of motion of multibody systems either in ODE or in DAE form. Therefore, it is easier to implement this method in a general purpose simulation software.

## 1 Introduction

The inverse dynamics analysis of multibody systems aims at determining the control inputs in order to track a prescribed trajectory, e.g. an end-effector trajectory. Inverse dynamics is an important tool for the development of efficient feed-forward control strategies. For fully actuated multibody systems, this is rather straightforward, and yields a purely algebraical expression for the feed-forward control, see e.g. Ref. [24]. In contrast, the inverse dynamics problem appears particularly difficult to solve for underactuated multibody systems, where there are less control inputs than degrees of freedom. Typical sources of underactuation include the presence of passive joints and body-flexibility. One challenge in feed-forward control design of these underactuated multibody systems is that the inverse model might possess internal dynamics, which might be stable or unstable. For inverse dynamics of underactuated multibody systems Ref. [7] proposes to supplement the equations of motion with the trajectory constraints, also called servo constraints, and to solve the resulting set of differential-algebraic equations (DAEs) for given initial values using a time integration method. However, this approach is limited to so-called differential flat systems [7] and minimum phase systems [19], i.e. systems without or with stable internal dynamics, respectively. In contrast non-minimum phase systems have unbounded internal dynamics and forward time integration would yield unbounded states

and control inputs. Also for non-minimum phase systems nonlinear control techniques such as feedback linearization cannot be used. A detailed analysis of minimum and non-minimum phase systems can be found in [15, 18].

This paper addresses the inverse dynamics of non-minimum phase underactuated multi-body systems, which represent an important class of practical applications. For example, non-minimum phase behavior can occur when joint actuators are used to control the end effector of a flexible manipulator [23] or a manipulator with passive joints [20]. The inverse dynamics of a non-minimum phase system is known to be non-causal, which means that the control forces and torques should start before the beginning of the trajectory (pre-actuation phase) and continue after the end-point is reached (post-actuation phase).

In order to address this problem, Ref. [22] presents a methodology for underactuated multibody systems in minimal coordinates based on the explicit derivation of the equations of the internal dynamics using a diffeomorphic coordinate transformation. Even though the internal dynamics is unstable, numerical solvers for boundary value problems allow to compute a bounded solution, including pre- and post-actuation phases. This so-called stable inversion solution approach [11, 25] originates in general nonlinear non-minimum phase systems. It is worth mentioning that no algebraic constraint appears in the equations of the internal dynamics with this approach.

Another approach is proposed here which does not require the explicit derivation of the internal dynamics equations. Instead, the method is based on supplementing the equations of motion of the multibody system by a set of trajectory constraints. The methodology is thus more suitable for implementation in an existing simulation code. The main difference compared to the work in Refs. [7, 19] is that the resulting problem cannot be solved as an initial value problem by time integration, and therefore an optimal control problem is derived. Thanks to this important difference, the methodology is no more limited to differentially flat or minimum phase systems but it can be exploited to compute the non-causal inverse dynamics of non-minimum phase systems.

The methodology can be applied regardless of the particular formulation of the equations of motion which can be written either as an ODE in case of a minimal coordinates formulation or as a DAE otherwise. In this paper minimal and absolute coordinate formulations are explored for the simulation of multibody systems. The generalized- $\alpha$  method based on the Newmark formulae is considered for time discretization. Based on a direct transcription of the optimal control problem, a robust direct collocation method is used, which leads to a large but sparse nonlinear programming problem. The unknowns of the optimization problem are the displacements, velocities, accelerations, Lagrange multipliers and control inputs at each time step, whereas the equations of motion, the time integration formulae and the trajectory constraints are treated as equality constraints. The sparse gradients of the optimization constraints are computed using a semi-analytical method. Compared to a finite difference approach, this method improves the convergence of the optimization algorithm and reduces considerably the computational cost. In order to illustrate the methodology, serial and parallel manipulators with passive joints are considered.

## 2 OPTIMAL CONTROL METHOD FOR MULTIBODY SYSTEMS

Figure 1 shows the typical block diagram of a controlled dynamic system, with a feed-forward control and a feedback control part. The feed-forward control provides the necessary control input  $\mathbf{u}_{ff}(t)$  such that the nominal system without disturbances and uncertainties follows precisely a given output trajectory. A feedback controller providing  $\mathbf{u}_{fb}(t)$  is used to ensure robust dynamic performance. Optimal control methods are often used to determine the motion and feed-forward control inputs  $\mathbf{u}_{ff}(t)$  for working point changes. Thereby the motion of the system output is not pre-described and transitions between working points in minimal time or with

minimal energy can be obtained. In this paper such an optimal control algorithm is extended to feed-forward control design for exact trajectory tracking of underactuated multibody systems. In the following, the feedback term is not considered, such that  $\mathbf{u}(\mathbf{t}) = \mathbf{u}_{ff}(\mathbf{t})$ .

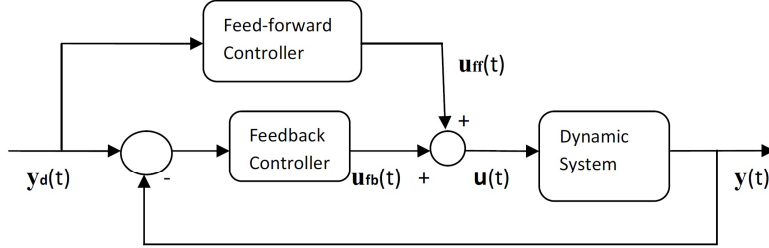


Figure 1: Block diagram: Dynamic system with feed-forward and feedback controller

## 2.1 DYNAMICS OF MULTIBODY SYSTEMS

Absolute coordinates [13] and minimal coordinates [14] are frequently used for the simulation of multibody systems. Both types of formulation are considered here for optimal control of underactuated multibody systems. The equations of motion of a multibody system with trajectory constraints take the general form

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{g}(\mathbf{q}, \dot{\mathbf{q}}, t) + \mathbf{B}^T(\mathbf{q})\boldsymbol{\lambda} - \mathbf{A}\mathbf{u} = \mathbf{0} \quad (1)$$

$$\boldsymbol{\Phi}(\mathbf{q}) = \mathbf{0} \quad (2)$$

$$\mathbf{y}(\mathbf{q}) - \mathbf{y}_d(t) = \mathbf{0} \quad (3)$$

where  $\mathbf{q}$  is the vector of  $r$  coordinates,  $\mathbf{M}$  is the mass matrix,  $\mathbf{g}$  is the vector of internal and complementary inertia forces,  $\boldsymbol{\Phi}$  is the vector of  $m$  kinematic constraints,  $\boldsymbol{\lambda}$  is the vector of  $m$  Lagrange multipliers and  $\mathbf{B} = \partial\boldsymbol{\Phi}(\mathbf{q})/\partial\mathbf{q}$  is the matrix of constraint gradients. The input matrix  $\mathbf{A}$  distributes the  $s$  control inputs  $\mathbf{u}$  onto the directions of the system coordinates, whereby  $\mathbf{A}$  is often a constant matrix. The  $s$  trajectory constraints are represented by Eq. (3), where the operator  $\mathbf{y}(\mathbf{q})$  computes the outputs of the dynamic system and the function  $\mathbf{y}_d(t)$  represents the prescribed path. A typical example for the system output is the end-effector point of a manipulator. The case of a minimal coordinate formulation is a particular case of Eqs. (1-3) with no Lagrange multiplier and no kinematic constraint. For an underactuated systems with  $r$  coordinates,  $m$  kinematic constraints and  $s$  trajectory constraints, we have  $r - m - s > 0$ . In addition this formalism can be generalized to account for design parameters in the optimization process, as shown in Ref. [3].

## 2.2 Optimization problem

Optimal control aims at the optimal choice of inputs for a dynamic system. In order to formulate the optimal control problem, an objective function with a general *Bolza* form can be considered:

$$J = \int_{t_i}^{t_f} L(\mathbf{q}(t), \dot{\mathbf{q}}(t), \boldsymbol{\lambda}(t), \mathbf{u}(t), t) dt + E(\mathbf{q}(t_f), \dot{\mathbf{q}}(t_f)) \quad (4)$$

This objective function combines a cost integral over the interval  $[t_i, t_f]$  (*Lagrange term*) and a function of the state variables at the final time  $t_f$  (*Mayer term*).

In general, path constraints and termination constraints impose restrictions to the dynamics of the system. This can yield to inequalities constraints or equalities constraints. In this

paper no termination or inequality constraints is considered but Eqs. (1-3) are treated as path equality constraints which are written in compact form as

$$\mathbf{c}(\mathbf{q}(t), \dot{\mathbf{q}}(t), \ddot{\mathbf{q}}(t), \boldsymbol{\lambda}(t), \mathbf{u}(t)) = \mathbf{0} \quad (5)$$

Optimal control methods are generally classified into: (i) dynamic programming methods [5], which require the solution of a partial differential equation; (ii) indirect methods [2], which require the solution of a boundary value problem; (iii) direct methods, which transform the optimal control problem into a nonlinear programming problem.

Direct methods formulate the optimality condition after time discretization of the problem, and they offer several advantages such as robustness, an easy reformulation for different problems and the ability to solve complex and unstable nonlinear problems. Among direct methods, a further distinction can be done between direct transcription methods [6], single shooting methods and multiple shooting methods [12]. This paper focus on the direct transcription method where the time integration and optimization problems are solved simultaneously. This means that all control and state variables at each step of the time grid are treated as optimization variables, leading to a large but sparse nonlinear programming (NLP) problem.

### 2.3 Direct transcription method

In the direct transcription method, the optimal control problem is discretized in time with  $N$  grid points  $t^{(k)}$ ,  $k = 1, \dots, N$ . The set of design variables includes the displacements  $\mathbf{q}$ , velocities  $\dot{\mathbf{q}}$ , accelerations  $\ddot{\mathbf{q}}$ , Lagrange multipliers  $\boldsymbol{\lambda}$  and control inputs  $\mathbf{u}$  of the system at each time step. For a dynamic system represented by Eqs. (1-3), the generalized- $\alpha$  method Refs. [10, 1], that is based on the Newmark formulae Ref. [17] can be used to discretize the problem in time. This provides

$$\dot{\mathbf{q}}^{(k+1)} = \dot{\mathbf{q}}^{(k)} + (1 - \gamma)h\mathbf{a}^{(k)} + \gamma h\mathbf{a}^{(k+1)} \quad (6)$$

$$\mathbf{q}^{(k+1)} = \mathbf{q}^{(k)} + h\dot{\mathbf{q}}^{(k)} + \left(\frac{1}{2} - \beta\right)h^2\mathbf{a}^{(k)} + \beta h^2\mathbf{a}^{(k+1)} \quad (7)$$

with an additional acceleration update formulae

$$\mathbf{a}^{(k+1)} = \frac{1}{(1 - \alpha_m)}[-\alpha_m\mathbf{a}^{(k)} + (1 - \alpha_f)\ddot{\mathbf{q}}^{(k+1)} + \alpha_f\ddot{\mathbf{q}}^{(k)}] \quad (8)$$

where  $\gamma$ ,  $\beta$ ,  $\alpha_m$  and  $\alpha_f$  are numerical parameters of the generalized- $\alpha$  method and can be chosen to combine unconditional stability and second-order accuracy. The numerical damping can be adjusted to obtain a given spectral radius at infinite frequencies  $\rho_\infty$ , see Ref. [10]. However, others integrations methods could be used, e.g. trapezoidal rule.

Equations (6-8) leads to the set of design variables,

$$\mathbf{x} = (\mathbf{q}^{(1)}; \dot{\mathbf{q}}^{(1)}; \ddot{\mathbf{q}}^{(1)}; \mathbf{a}^{(1)}; \boldsymbol{\lambda}^{(1)}; \mathbf{u}^{(1)}; \dots; \mathbf{q}^{(N)}; \dot{\mathbf{q}}^{(N)}; \ddot{\mathbf{q}}^{(N)}; \mathbf{a}^{(N)}; \boldsymbol{\lambda}^{(N)}; \mathbf{u}^{(N)}). \quad (9)$$

The dimension of  $\mathbf{x}$  is  $n = N(4r + m + s)$ , which is generally large in practical applications. The design domain is defined as  $D \subset \mathbb{R}^n$ . Additional parameters can also be included in the set of design variables to be optimized, e.g. to represent the structural design of the mechanical system.

The direct transcription method can treat inequality and equality constraints easily, which is an advantage compared to other approaches. If the path equality and inequality constraints are imposed only at the nodes of the time grid, the optimization problem is restated as a nonlinear programming (NLP) problem. In the present case with only path equality constraints, the NLP problem takes the general form

$$\begin{aligned} \min \quad & J^d(\mathbf{x}) \\ \text{s.t} \quad & \mathbf{c}^d(\mathbf{x}) = \mathbf{0} \end{aligned} \quad (10)$$

where  $J^d : D \subset \mathbb{R}^n \rightarrow \mathbb{R}$  is the discretized objective function and  $\mathbf{c}^d : D \subset \mathbb{R}^n \rightarrow \mathbb{R}^{n_{ec}}$  is the vector of discretized path equality constraints that includes the time integration formulae.

The optimal solution satisfies the Karush-Kuhn-Tucker necessary conditions of optimality. Standard NLP solvers can be used to solve such problems efficiently, see Ref. [4]. For instance, Sequential Quadratic Programming methods (SQP), see Ref. [9], are iterative methods which solve at each iteration a quadratic programming problem.

The discretization of the objective function in Eq. (4) according to the Euler explicit method leads to

$$J^d = \sum_{k=1}^{N-1} L(\mathbf{q}^k, \dot{\mathbf{q}}^k, \boldsymbol{\lambda}^k, \mathbf{u}^k) h + E(\mathbf{q}^N, \dot{\mathbf{q}}^N) \quad (11)$$

where  $h$  is the time step of the discretization. In order to increase the accuracy a higher order discretization could be used such as the trapezoidal rule.

At each time step  $k = 1, \dots, N$ , the equations of motion, Eqs. (1-3), are treated as equality constraints

$$\mathbf{c}_{\mathbf{m}}^{(k)} = \begin{bmatrix} \mathbf{M}(\mathbf{q}^{(k)})\ddot{\mathbf{q}}^{(k)} + \mathbf{g}(\mathbf{q}^{(k)}, \dot{\mathbf{q}}^{(k)}, t^{(k)}) + \mathbf{B}^T(\mathbf{q}^{(k)})\boldsymbol{\lambda}^{(k)} - \mathbf{A}\mathbf{u}^{(k)} \\ \Phi(\mathbf{q}^{(k)}) \\ \mathbf{y}(\mathbf{q}^{(k)}) - \mathbf{y}_d(t^{(k)}) \end{bmatrix} = \mathbf{0} \quad (12)$$

Similarly, the integration formulae of the generalized- $\alpha$  method, Eqs. (6-8), are also implemented as equality constraints for  $k = 1, \dots, N-1$  to ensure continuity of the solution and consistency between displacement, velocity and acceleration variables

$$\mathbf{c}_{\alpha}^{(k)} = \begin{bmatrix} \dot{\mathbf{q}}^{(k+1)} - \dot{\mathbf{q}}^{(k)} - (1 - \gamma)h\mathbf{a}^{(k)} - \gamma h\mathbf{a}^{(k+1)} \\ \mathbf{q}^{(k+1)} - \mathbf{q}^{(k)} - h\dot{\mathbf{q}}^{(k)} - (\frac{1}{2} - \beta)h^2\mathbf{a}^{(k)} - \beta h^2\mathbf{a}^{(k+1)} \\ (1 - \alpha_m)\mathbf{a}^{(k+1)} + \alpha_m\mathbf{a}^{(k)} - (1 - \alpha_f)\ddot{\mathbf{q}}^{(k+1)} - \alpha_f\ddot{\mathbf{q}}^{(k)} \end{bmatrix} = \mathbf{0} \quad (13)$$

The vector of equality constraints then comprises each component of these constraints at each time step:

$$\mathbf{c}^d = (\mathbf{c}_{\mathbf{m}}^{(1)}; \mathbf{c}_{\alpha}^{(1)}; \dots; \mathbf{c}_{\mathbf{m}}^{(N-1)}; \mathbf{c}_{\alpha}^{(N-1)}; \mathbf{c}_{\mathbf{m}}^{(N)}) = \mathbf{0}. \quad (14)$$

In the above Eqs. (12-13), each constraint only depends on a few components of the large vector  $\mathbf{x}$  given by Eq. (9). For this reason, the resulting NLP problem has a very sparse structure.

## 2.4 Semi-analytical sensitivity analysis

Most deterministic optimization algorithms require the evaluation of the gradients of the objective function and design constraints with respect to all design variables. Those gradients may be obtained by finite difference approximation; however, if the particular sparse structure of the optimal control problem is not exploited, this will lead to rather inefficient computations. Instead of finite difference techniques, semi-analytical methods are used here to compute the sensitivities of the design functions  $J^d$  and  $\mathbf{c}^d$  with respect to the design variables  $\mathbf{x}$ . For example, direct differentiation methods described in Ref. [8] can be used in the present context. Those semi-analytical methods lead to more accurate estimation of the gradients which improves the convergence speed of the optimization algorithm. And they naturally exploit the sparse structure of the NLP problem.

For the objective function the discretized *Bolza* form in Eq. (11) has as Jacobian matrix

$$\frac{\partial J^d}{\partial \mathbf{x}} = h \left( \frac{\partial L}{\partial \mathbf{q}^{(1)}}; \frac{\partial L}{\partial \dot{\mathbf{q}}^{(1)}}; \mathbf{0}; \mathbf{0}; \frac{\partial L}{\partial \boldsymbol{\lambda}^1}; \frac{\partial L}{\partial \mathbf{u}^{(1)}}; \dots; \right. \\ \left. \frac{\partial L}{\partial \mathbf{q}^{(N-1)}}; \frac{\partial L}{\partial \dot{\mathbf{q}}^{(N-1)}}; \mathbf{0}; \mathbf{0}; \frac{\partial L}{\partial \boldsymbol{\lambda}^{(N-1)}}; \frac{\partial L}{\partial \mathbf{u}^{(N-1)}}; \frac{1}{h} \frac{\partial E}{\partial \mathbf{q}^{(N)}}; \frac{1}{h} \frac{\partial E}{\partial \dot{\mathbf{q}}^{(N)}}; \mathbf{0}; \mathbf{0}; \mathbf{0}; \mathbf{0} \right) \quad (15)$$

The gradient of the equality constraints is based on the gradient of Eqs. (12-13). Introducing the subset of design variable in each time step  $k$

$$\mathbf{x}_k = (\mathbf{q}^{(k)}; \dot{\mathbf{q}}^{(k)}; \ddot{\mathbf{q}}^{(k)}; \mathbf{a}^{(k)}; \boldsymbol{\lambda}^{(k)}; \mathbf{u}^{(k)}) \quad (16)$$

we can write

$$\frac{\partial \mathbf{c}_m^{(k)}}{\partial \mathbf{x}_k} = \begin{pmatrix} \mathbf{K}_T^{(k)} & \mathbf{C}_T^{(k)} & \mathbf{M}^{(k)} & \mathbf{0} & \mathbf{B}^T^{(k)} & -\mathbf{A} \\ \mathbf{B}^{(k)} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \frac{\partial \mathbf{y}^{(k)}}{\partial \mathbf{q}^{(k)}} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{pmatrix}, \quad (17)$$

where  $\mathbf{K}_T^{(k)}$  is the matrix associated with the position dependent forces and  $\mathbf{C}_T^{(k)}$  is associated with the velocity dependent forces. In the case of the absolute coordinate formulation these are the tangential stiffness and damping matrices, respectively, at each time step  $t^{(k)}$ . Observing that Eq. (13) is linear, its differentiation leads to constant matrices which depend on the numerical parameters of the generalized- $\alpha$  method

$$\frac{\partial \mathbf{c}_\alpha^{(k)}}{\partial \mathbf{x}_k} = \begin{pmatrix} \mathbf{0} & -\mathbf{I} & \mathbf{0} & -(1-\gamma)h\mathbf{I} & \mathbf{0} & \mathbf{0} \\ -\mathbf{I} & -h\mathbf{I} & \mathbf{0} & -(\frac{1}{2}-\beta)h^2\mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & -\alpha_f\mathbf{I} & \alpha_m\mathbf{I} & \mathbf{0} & \mathbf{0} \end{pmatrix} = \mathbf{C}_{\alpha,0} \quad (18)$$

and

$$\frac{\partial \mathbf{c}_\alpha^{(k)}}{\partial \mathbf{x}_{k+1}} = \begin{pmatrix} \mathbf{0} & \mathbf{I} & \mathbf{0} & -\gamma h\mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{I} & \mathbf{0} & \mathbf{0} & -\beta h^2\mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & -(1-\alpha_f)\mathbf{I} & (1-\alpha_m)\mathbf{I} & \mathbf{0} & \mathbf{0} \end{pmatrix} = \mathbf{C}_{\alpha,1} \quad (19)$$

With these elements the Jacobian matrix of the constraints Eqs. (12-13) is obtained as

$$\frac{\partial \mathbf{c}_d}{\partial \mathbf{x}} = \begin{pmatrix} \frac{\partial \mathbf{c}_m^{(1)}}{\partial \mathbf{x}_1} & & & & & \\ \mathbf{C}_{\alpha,0} & \mathbf{C}_{\alpha,1} & & & & \\ & \frac{\partial \mathbf{c}_m^{(2)}}{\partial \mathbf{x}_2} & & & & \\ & \mathbf{C}_{\alpha,0} & \mathbf{C}_{\alpha,1} & & & \\ & & & \ddots & & \\ & & & & \frac{\partial \mathbf{c}_m^{(N-1)}}{\partial \mathbf{x}_{N-1}} & \\ & & & & \mathbf{C}_{\alpha,0} & \mathbf{C}_{\alpha,1} \\ & & & & & \frac{\partial \mathbf{c}_m^{(N)}}{\partial \mathbf{x}_N} \end{pmatrix}. \quad (20)$$

We can note the sparsity and the size of this NLP problem shown in this Jacobian matrix and in its block diagonal structure. The size of the design variables and of the Jacobian matrix of the constraints and objective function increases strongly with the time grid discretization. An elimination of the linear constraints could be applied to reduce the number of design variables and constraints in the optimization process. For example, in our problem, we could eliminate the time integration constraints that are always linear, see Eq. (13), using a partitioning of the design variables. However, this may alter the sparsity pattern of the optimization problem, so that the benefit of such a constraint elimination might not be guaranteed. Therefore, we do not consider an elimination of the linear constraints in this paper.

### 3 APPLICATIONS - UNDERACTUATED MULTIBODY SYSTEMS

In this paper, the optimal control approach is applied for trajectory tracking of non-minimum phase underactuated multibody systems. In nonlinear control theory, a system is called minimum-

phase, if its internal dynamics, i.e. its zero-dynamics, is asymptotically stable [15], otherwise the system is non-minimum phase. Thus a minimum phase system has dynamics and inverse dynamics which are both causal and stable. A causal or non-anticipative is a system where the output depends on past or current inputs but never on future inputs. In contrast non-minimum phase systems might yield unbounded inverse dynamics, if solved by forward time integration of Eqs. (1-3). However, also bounded solution can be computed, e.g. by stable inversion [11] or the approach presented in this paper. In contrast, the bounded solution of inverse dynamics of a non-minimum phase system is known to be non-causal, which means that the control forces and torques should start before the beginning of the output trajectory (pre-actuation phase) and continue after the end-point is reached (post-actuation phase).

### 3.1 Underactuated serial manipulator with one passive joint

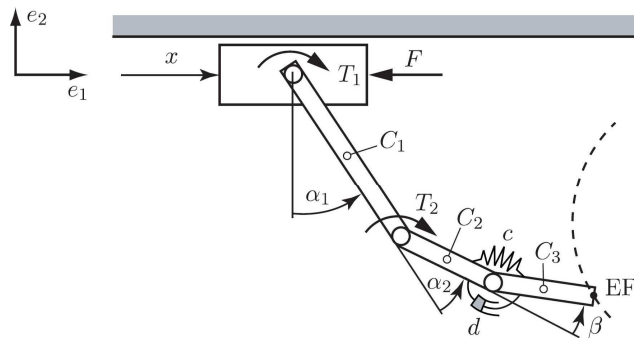


Figure 2: Planar serial manipulator with one passive joint

Here an underactuated manipulator with one passive joint and kinematic redundancy is considered as shown in Fig. 2. The manipulator moves in the horizontal plane and consists of a cart on which a chain of three arms is mounted. It is actuated by the control inputs

$$\mathbf{u} = (F, T_1, T_2)^T \quad (21)$$

where  $F$  is the force exerted on the car,  $T_1$  and  $T_2$  are the joint torques. The homogenous arms have lengths  $l_1$  and  $l_2 = l_3$ , mass  $m_1$  and  $m_2 = m_3$  and inertia  $I_1$  and  $I_2 = I_3$ . The third arm is connected by a passive joint to arm 2 which is supported by a parallel spring-damper combination with spring constant  $c$  and damping coefficient  $d$ . The physical parameters of the manipulator are summarized in Tab. 1.

cart	$m_c = 3 \text{ kg}$		
arm 1	$m_1 = 6.875 \text{ kg}$	$I_1 = 0.5743 \text{ kgm}^2$	$l_1 = 1.0 \text{ m}$
arm 2	$m_2 = 3.438 \text{ kg}$	$I_2 = 0.0723 \text{ kgm}^2$	$l_2 = 0.5 \text{ m}$
arm 3	$m_3 = 3.438 \text{ kg}$	$I_3 = 0.0723 \text{ kgm}^2$	$l_3 = 0.5 \text{ m}$
passive joint	$c = 50 \text{ Nm/rad}$	$d = 0.25 \text{ Nms/rad}$	

Table 1: Parameters of the underactuated serial manipulator.

The control goal is to force the cart and the end-effector to follow a predefined trajectory as closely as possible. The problem has three trajectory constraints that are imposed on the

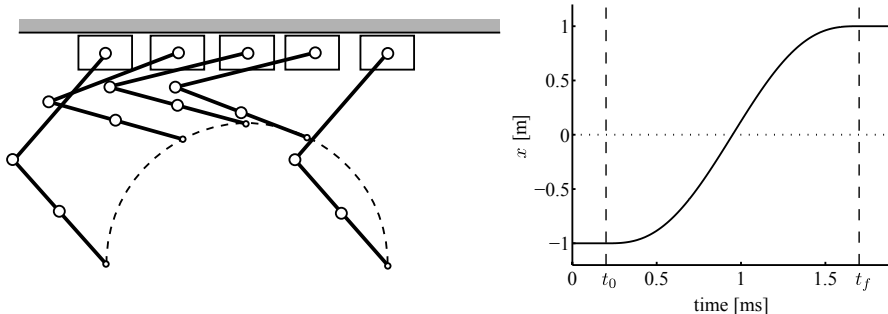


Figure 3: Underactuated serial manipulator: Desired trajectory

cart motion  $x$ , and on the end-effector positions  $(EF_x, EF_y)$ . In Eq. (12),  $\mathbf{y}(\mathbf{q}) = (x, EF_x, EF_y)$  describes the output of the manipulator, and  $\mathbf{y}_d$  is the desired output trajectory of the manipulator. The end-effector point should follow a half-circular trajectory. The center of the half-circle is at position  $(0.0, -1.5)$ [m] and the radius is 1.0 m. Also the cart should move from starting position  $-1.0$ m to the final position  $1.0$ m. The system has  $s = 3$  trajectory constraints. The trajectories are shown in Fig. 3. The end-effector point and the cart should follow the trajectory in the short time period of 1.5s, which describes an aggressive manoeuvre. The pre-actuation period is 0.2s and the post-actuation period is 0.3s. The design of the pre/post-actuation phases is an important step to bound well the inverse dynamic problem of non-minimum phase systems. Here, the knowledge of the time constants of the system's zero is crucial, see Ref. [16] for a more detailed discussion. A stable inversion [11, 25], which requires the solution of a two-sided boundary value problem by finite differences, was used in Ref. [22] for this problem. This methodology is based on the explicit diffeomorphic coordinate transformation of the equation of motion into the so-called nonlinear input-output normal-form, from which then the equations of the internal dynamics is derived. A detailed analysis is given in Refs. [22, 20]. In this example the  $\beta$  coordinate describes the internal dynamics, and in this case forward time integration would yield unbounded values for  $\beta$  and the control inputs  $\mathbf{u}$ . With stable inversion one obtains bounded but non-causal solution for output trajectory tracking, in other words the pre- and post-actuation phases are needed.

Using direct methods in optimal control we can take advantage of the trajectory knowledge of the system, see Fig. 3, and incorporate it as initial guess in the optimization process. We consider the motion of an equivalent rigid manipulator without deformation of the passive joint ( $\beta(t) = 0.0$ ), see Fig. 7. With that choice kinematic and trajectory constraints are already satisfied before the optimization process starts. A good initial guess is crucial to solve the problems presented in this paper.

We propose to solve this inverse dynamics problem using the optimal control approach based in the direct transcription method. We consider that the manipulator should realize its task with the minimum displacement of the passive joint. The objective function is then written as

$$J = \int_{t_i}^{t_f} \beta^2 dt \quad (22)$$

where  $\beta$  is the passive joint angle. Thus, the optimization goal is to minimize this objective function, so that a bounded solution of the unstable internal dynamics is enforced.

### 3.1.1 Minimal coordinates formulation

The manipulator shown in Fig. 2 described in minimal coordinates has the generalized coordinates  $\mathbf{q} = (x, \alpha_1, \alpha_2, \beta)^T$ . In this formulation the number of generalized coordinates is equal



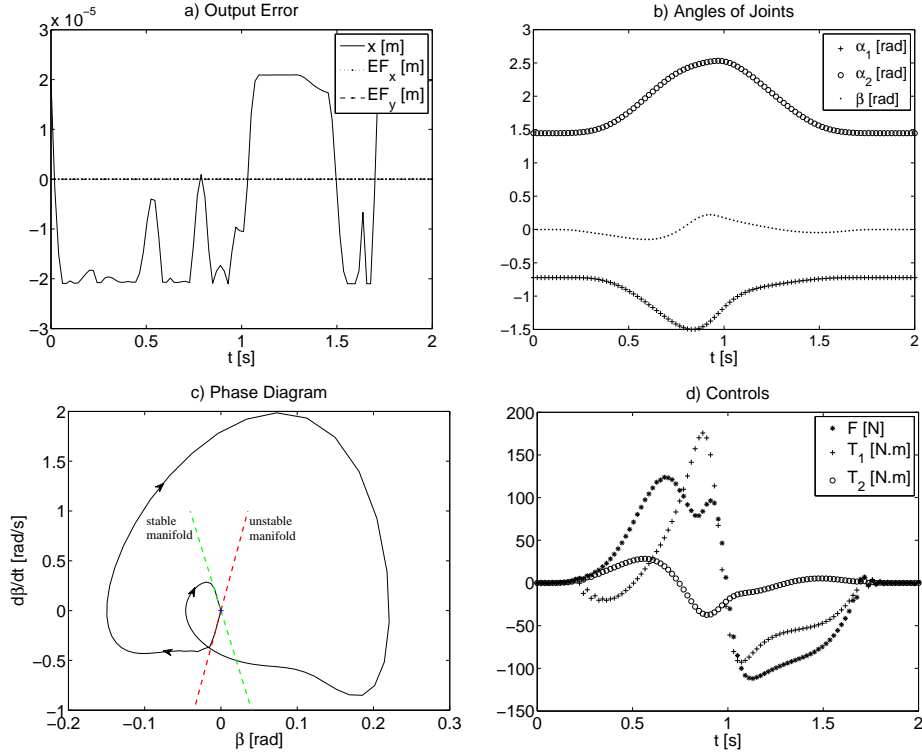


Figure 4: Results - Underactuated serial manipulator - Minimal coordinates

to the number of degrees of freedom. This is a particular case of Eqs. (1-3) without Lagrange multiplier and kinematic constraint. The constraints on the trajectory include a constraint on the tip of mechanism and on the cart displacement.

The results using this formulation with 100 time steps and  $\rho_\infty = 0.50$  are shown Fig. 4. The objective function value is  $J = 0.01560$  that is obtained after 6 iterations. Figure 4a shows the numerical error of the cart and end-effector trajectory constraints, which is an excellent result. The maximum position error is  $2.1 \cdot 10^{-5}$  m for the cart and nearly zero for the end-effector. It should be noted that the trajectory constraints are nonlinear equations, in terms of minimal coordinates. Figure 4b shows the angular displacements of the manipulator arms. The initial and final values before and after the pre- and post-actuation are the same, as shown schematically in Fig. 3. Figure 4c shows the phase diagram of the driven internal dynamics which has fixed point  $(\beta, \dot{\beta}) = (0.0, 0.0)$ . This point is a saddle point, that is an unstable equilibrium point. The unstable and stable manifolds pass by this equilibrium point. The trajectory of  $\beta$  starts at time  $t_0$  on the unstable manifold and ends at time  $t_f$  on the stable manifold, providing a pre- and post-actuation phase. This is in accordance with the stable inversion method in Ref. [11], which is applied to this manipulator in Ref. [22]. Figure 4d shows the computed control of the system, which are obviously bounded. The pre- and post-actuation phase can be observed on all graphics in Fig. 4.

### 3.1.2 Absolute coordinates formulation

The manipulator studied in the previous section using minimal coordinates is now described using the absolute coordinates of some nodes of the system, as shown in Fig. 5, with

$$\mathbf{q} = (x_0, x_1, y_1, \theta_1, x_2, y_2, x_3, y_3, \theta_3, x_4, y_4, x_5, y_5, \theta_5, x_6, y_6)^T, \quad (23)$$

where  $x_i$  and  $y_i$  are the translations of node  $i$  along axes  $e_1$  and  $e_2$ , respectively, and  $\theta_i$  is the absolute orientation angle of node  $i$ . Those  $r = 16$  coordinates are not independent but they have to satisfy  $m = 12$  kinematic constraints. This description is consistent with the finite element approach for flexible systems described in [3], whereby constraints impose the rigidity of the system. Compared to minimal coordinates, the absolute coordinates lead to more equations to solve and more generalized coordinates, but the equations are simpler and have a sparse structure. The nonlinear equations of motion using absolute coordinates have the

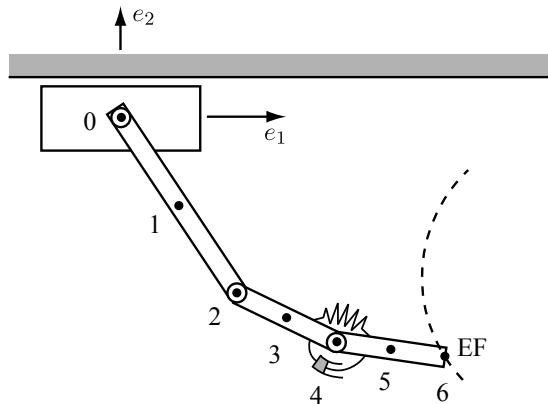


Figure 5: Underactuated serial manipulator: Discretization in absolute coordinates

structure of Eqs. (1-3).

The results using this formulation with 100 time steps and  $\rho_\infty = 0.50$  are shown in Fig. 6, and they are similar to the results obtained using minimal coordinates. The final objective function value  $J = 0.01547$  is obtained after 4 iterations. Figure 6a shows the output error of the planar manipulator with passive joints using the optimal control approach with absolute coordinates. The maximum numerical error of the cart and end-effector is of the order of  $10^{-16}$ , which is a very satisfactory result. The trajectory constraints using absolute coordinates is completely linear, an advantage in comparison to minimal coordinates. In Eq. (12),  $\mathbf{y}(\mathbf{q}) = \mathbf{D}\mathbf{q} = (x_0, x_6, y_6)^T$  where the constant output matrix  $\mathbf{D}$  distributes the outputs onto the directions of the systems coordinates.

Figure 7 shows the evolution of the nonlinear optimization process for that rigid serial manipulator discretized in absolute coordinates. We can see a fast convergence, and the first iteration is already very close to the solution. Figure 7a represents the evaluation of the integrand of the objective function Eq. (22). Figure 7b represents the evaluation of the phase diagram.

### 3.1.3 Comparisons of results

The phase diagram of the internal dynamics as shown in Fig. 4c allows to study the global behavior of the nonlinear dynamics of the system. This is used to perform some comparative studies. Figure 8 compares the optimal control approach in absolute coordinates and minimal coordinates. In addition, the result using the nonlinear programming (NLP) method Refs. [4, 21] and a stable inversion method Ref. [22] are displayed. For these latter two methods, the internal dynamics of the manipulator is derived from symbolic calculations as single ordinary differential equation (ODE) of form

$$M_{int}(\beta)\ddot{\beta} = RHS_{int}(\beta, \dot{\beta}), \quad (24)$$

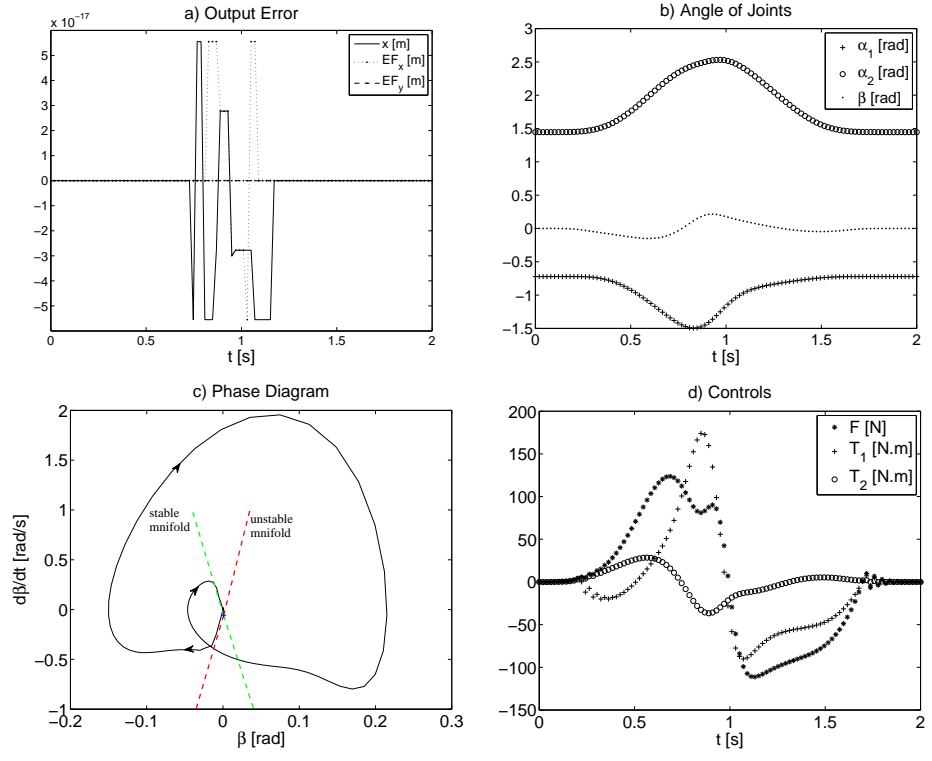


Figure 6: Results - Underactuated serial manipulator - Absolute coordinates

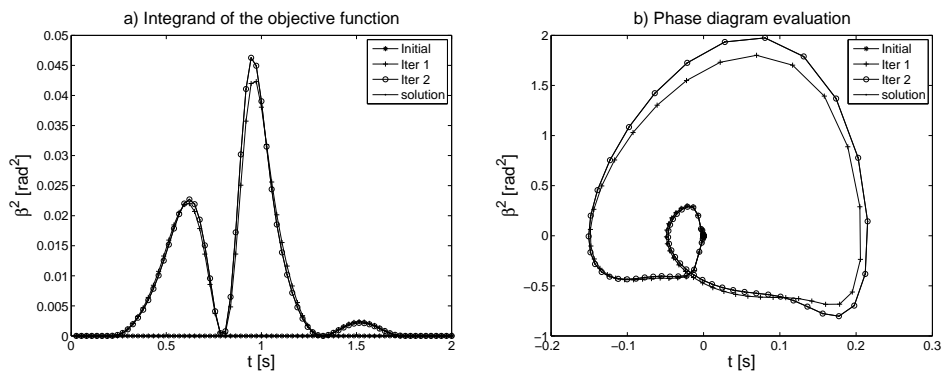


Figure 7: Iterative process of that nonlinear optimization.

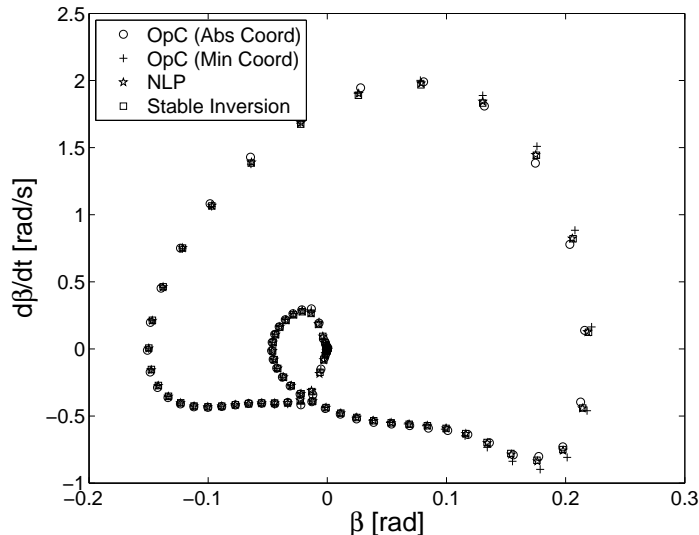


Figure 8: Phase diagram - Comparison of methods to solve problems of unstable internal dynamics.

see [22] for details. The comparison in Fig. 8 proves that all four approaches show very good agreement. However, it should be noted that the stable inversion method in Ref. [22] is not able to solve differential algebraic equations. Our results show that the optimal control approach proposed for inverse dynamics does not have this limitation, i.e. it is able to solve a more general class of problems than the stable inversion method.

The second comparison studies presented in Fig. 9 show the influence of the spectral radius at infinite frequencies  $\rho_\infty$ , that represents the amount of numerical damping in the generalized- $\alpha$  method for the time integration. Thereby  $\rho_\infty = 1.0$  means no numerical damping and  $\rho_\infty = 0.0$  means maximum numerical damping. For  $\rho_\infty = 1.0$  and  $\rho_\infty = 0.5$  nearly identical results are obtained, while for the maximal damped case with  $\rho_\infty = 0.0$  some deviations are obtained. The numerical damping is especially needed to stabilize the solution on the saddle point  $(\beta, \dot{\beta}) = (0.0, 0.0)$ , when the parameters of the parallel spring-damper combination are smaller, see in Fig. 10.

Figure 10 illustrates the influence of the spring coefficient  $c$  and damping coefficient  $d$  on the solution of the system. As expected, the motion of the  $\beta$  coordinate gets larger when weakening the spring-damper combination. Let us mention that, for the undamped system  $d = 0.0 \text{ Nms/rad}$ , we were not able to compute a solution for  $c < 18.0 \text{ Nm/rad}$ .

Finally, Fig. 11 shows the influence of the time step on the control input  $T_1$  with a spectral radius at infinite frequencies  $\rho_\infty = 0.5$ . The reference used is the stable inversion method with 150 time steps, see Ref. [22]. Some numerical oscillations can be observed but they tend to decrease when the time step is reduced. The post-actuation phase is the most sensitive to the time step and the oscillations are more important there. However, this is less critical, since after reaching the final point, one might consider to switch to a linear controller to regulate the stationary point.

### 3.2 Underactuated parallel manipulator with passive joint

A 3-RPR parallel manipulator [26] with one passive joint is now considered, see Fig. 12. That machine is composed of three prismatic joints, a triangular rigid plate, a rigid body and a passive joint with a spring-damper combination. The manipulator moves in the horizontal plane and

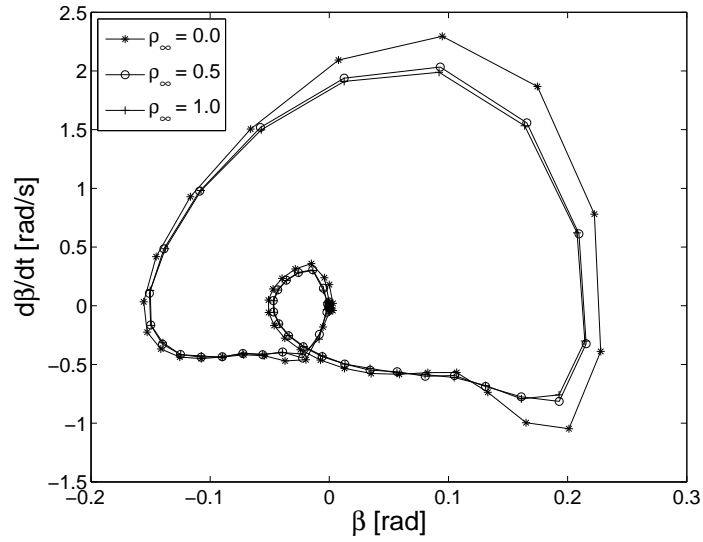


Figure 9: Phase diagram - Different values of numerical damping for the time integration.

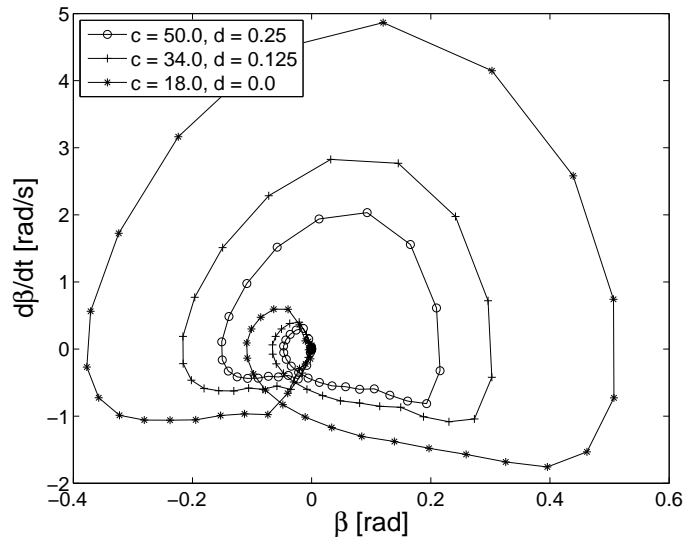


Figure 10: Phase diagram - Different values of spring-damper parameters:  $c$  [Nm/rad] and  $d$  [Nms/rad] ( $\rho_\infty = 0.5$ ).

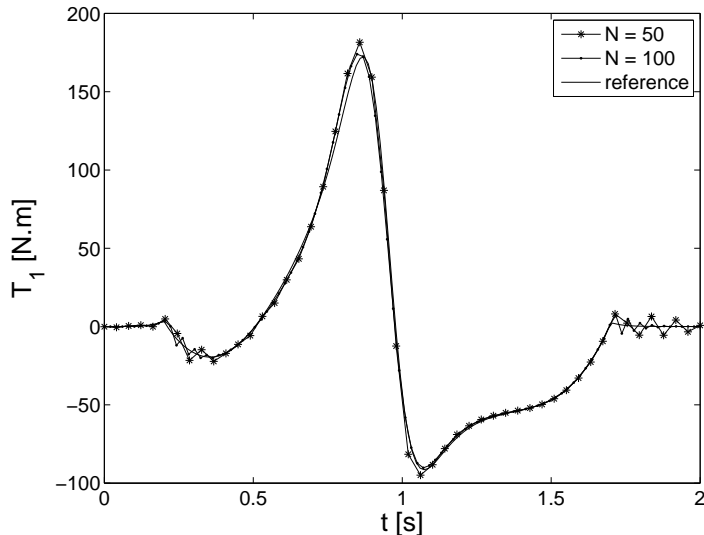


Figure 11: Time step influence in the control  $T_1$  ( $\rho_\infty = 0.5$ ).

the plate center is the end-effector. It is actuated by the control inputs

$$\mathbf{u} = (T_1, T_2, T_3)^T \quad (25)$$

where  $T_1$ ,  $T_2$  and  $T_3$  are the joint torques. The initial lengths of the prismatic joints are  $l_1 = l_3$  and  $l_2$ . The rigid body has length  $l_{rig}$ , mass  $m_{rig}$  and rotational inertia  $I_{rig}$ . The plate has an equilateral triangular geometry with side  $a$ , mass  $m$  and rotational inertia  $I$ . The parameters of the passive joint are: stiffness  $c$  and damping  $d$ . The physical parameters of this parallel manipulator are summarized in Tab. 2. The output of the system is the end-effector position

prismatic joint 1	$l_1 = 1.35$ m		
prismatic joint 2	$l_2 = 0.75$ m		
prismatic joint 3	$l_3 = 1.35$ m		
plate	$a = 0.5$ m	$m = 3.0$ kg	$I = 0.50$ kgm <sup>2</sup>
rigid link	$l_{rig} = 0.6$ m	$m_{rig} = 26.4$ kg	$I_{rig} = 0.80$ kgm <sup>2</sup>
passive joint	$c = 100$ Nm/rad	$d = 7.0$ Nms/rad	

Table 2: Parameters of the underactuated parallel manipulator.

and the plate orientation.

$$\mathbf{y} = (EF_x, EF_y, \theta_{plate})^T \quad (26)$$

In Eq. (12),  $\mathbf{y}(\mathbf{q}) = \mathbf{D}\mathbf{q} = (x_3, y_3, \theta_3)$  is a linear equation and describes the output of the manipulator, see Fig. 14, and  $\mathbf{y}_d$  is the desired output. The plate center should follow a half-circular trajectory with radius of 0.1 m) and the plate orientation should preserve its initial value  $\theta_{plate} = 0.0$  for all time, see Fig. 13. There are  $s = 3$  trajectory constraints. The output of the system should follow the trajectory in the short time period of 2.0s, which describes an aggressive manoeuver. The pre- and post-actuation period is 0.75s The optimal control approach is used to compute the inverse dynamics of this underactuated parallel manipulator with trajectory constraints. Since this manipulator with end-effector position as system output is non-minimum phase, the forward integration of Eqs. (1-3) would again yield unbounded coordinates and control inputs. The manipulator should realize its task with the minimum

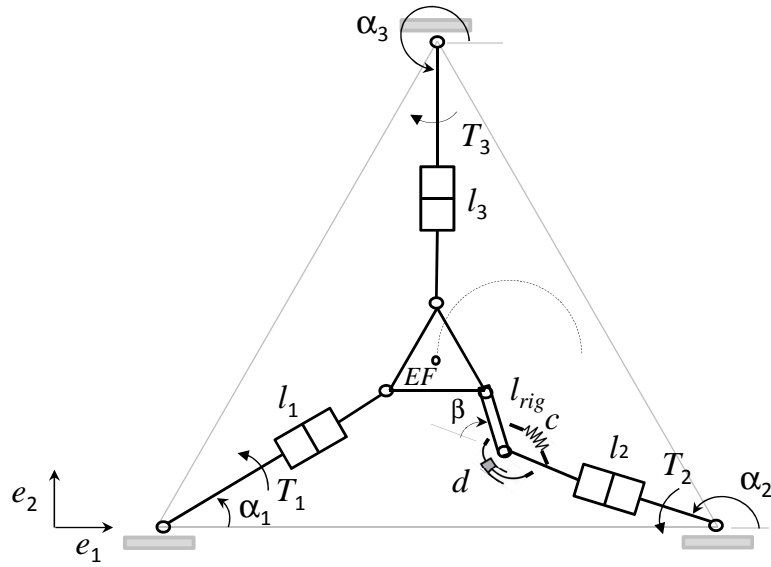


Figure 12: Underactuated parallel manipulator with passive joint

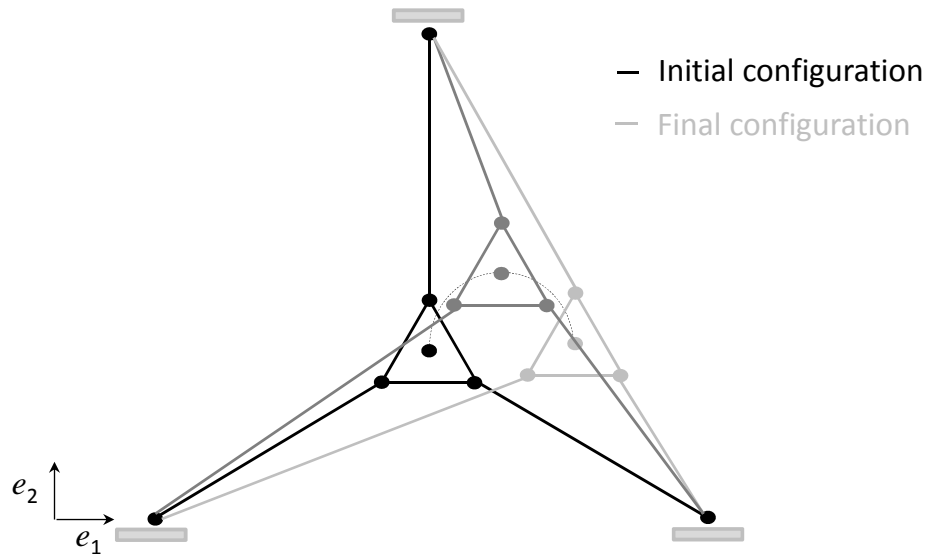


Figure 13: Underactuated parallel manipulator: Desired trajectory

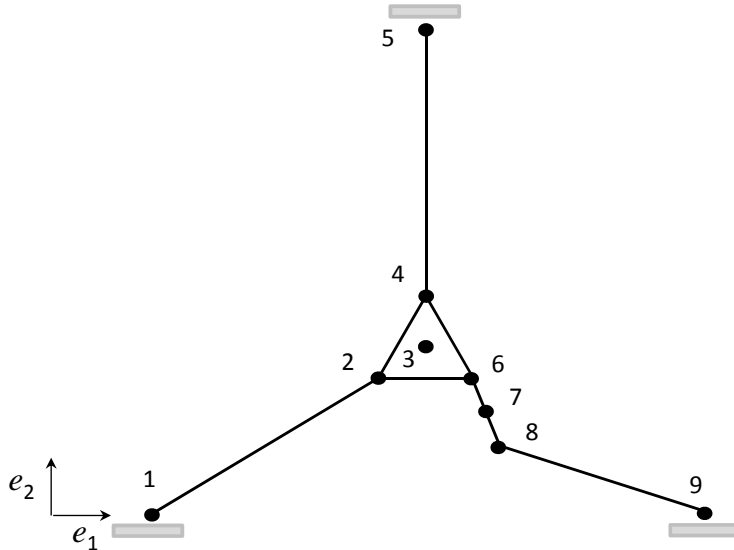


Figure 14: Underactuated parallel manipulator: Discretization in absolute coordinates

deformations of the spring-damper in the passive joint. The objective function is again given by Eq. (22), where  $\beta$  angle is the angle of the passive joint. Thus, the optimization goal is to minimize this objective function, so that a bounded solution can be obtained. The underactuated parallel manipulator discretized in absolute coordinates is shown in Fig. 14, with  $r = 17$  generalized coordinates in position:

$$\mathbf{q} = (\theta_1, x_2, y_2, \theta_2, x_3, y_3, \theta_3, x_4, y_4, \theta_5, x_6, y_6, x_7, y_7, \theta_7, x_8, y_8, \theta_9)^T, \quad (27)$$

The results using the optimal control approach with absolute coordinates formulation, 70 time steps and  $\rho_\infty = 0.0$  are shown in Fig. 15, The final objective function value  $J = 1.2864 \text{ e-}3$  is obtained after 20 iterations.

Figure 15a shows the output trajectory error, which is a very satisfactory result. Figure 15b shows the angles trajectories of the prismatic joints for that system. Figure 15c shows the internal dynamics of the system represented in the phase diagram of the passive joint. We can see the initial motion on the unstable manifold that contains the pre-actuation phase and the final motion on the stable manifold that contains the post-actuation phase, returning to the origin point (critical point)  $(\beta, \dot{\beta}) = (0.0, 0.0)$ . Figure 15d shows the control inputs that have to be applied to the system to the output follow the prescribed trajectory.

This example demonstrates that the optimal control method can be applied to compute the inverse dynamics of non-minimum phase systems with parallel kinematics.

## 4 CONCLUSION

This paper studies the inverse dynamics of non-minimum phase underactuated multibody systems, e.g. for the control of underactuated serial and parallel manipulators. A solution strategy based on an optimal control approach has been proposed. More precisely, a direct transcription method has been developed using the generalized- $\alpha$  time discretization scheme. The sparse gradients of the optimization constraints are computed using a semi-analytical method, which improves the convergence and reduces considerably the computational cost. The method is able to deal with a general class of systems implemented using minimal coordinates and absolute coordinates. The method is illustrated for the inverse dynamics of a planar underactuated serial manipulator with rigid links and an underactuated parallel manipulator. An important



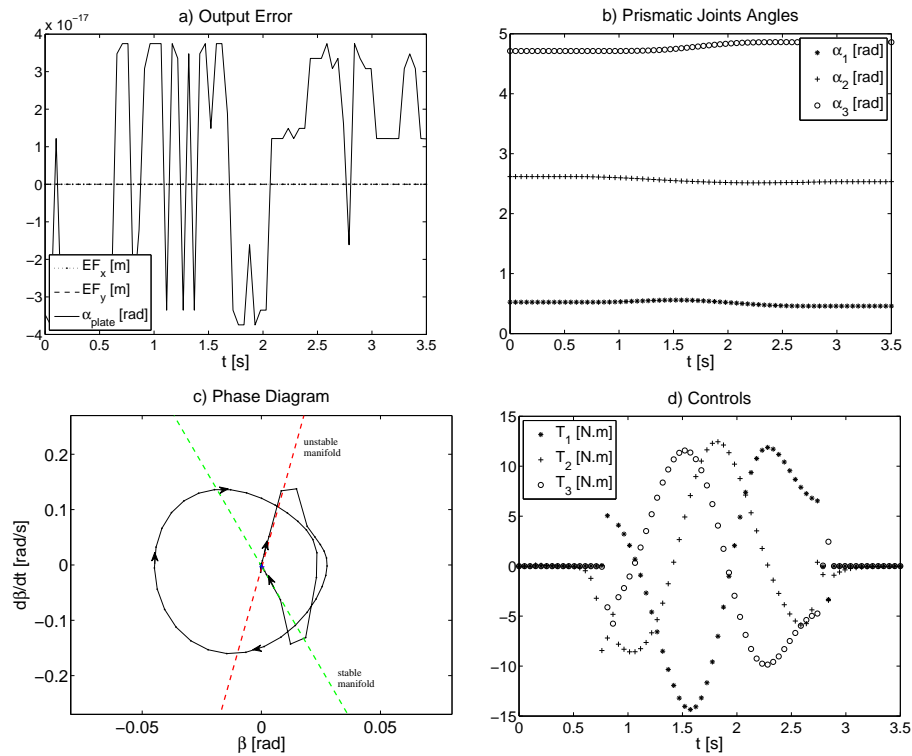


Figure 15: Results - Underactuated parallel manipulator - Absolute coordinates

advantage of the proposed approach is that it can be applied directly to the standard equations of motion of the multibody in DAE form. Therefore, the method can be easily implemented in a general purpose simulation software.

## References

- [1] M. Arnold and O. Brüls. Convergence of the generalized- $\alpha$  scheme for constrained mechanical systems. *Multibody System Dynamics*, 18(2):185–202, 2007.
- [2] V.L. Bakke. A maximum principle for an optimal control problem with integral constraints. *Journal of Optimization Theory and Applications*, 13(1):32–55, 1974.
- [3] G.J. Bastos and O. Brüls. Trajectory optimization of flexible robots using an optimal control approach. *Proceedings of the 1st Joint International Conference on Multibody System Dynamics*, 2010.
- [4] M.S. Bazaraa, H.D. Sherali, and C.M. Shetty. *Nonlinear Programming: Theory and Algorithms*. John Wiley & Sons, New Jersey, USA, 3 edition, 2006.
- [5] R. Bellman. *Dynamic Programming*. Princeton University Press, Princeton, N.J., USA, 2003.
- [6] J.T. Betts. *Practical Methods for Optimal Control and Estimation Using Nonlinear Programming*. Society for Industrial and Applied Mathematics, USA, 2 edition, 2010.

- [7] W. Blajer and K. Kolodziejczyk. A geometric approach to solving problems of control constraints: Theory and a DAE framework. *Multibody System Dynamics*, 11:343–364, 2004.
- [8] O. Brüls and P. Eberhard. Sensitivity analysis for dynamic mechanical systems with finite rotations. *International Journal for Numerical Methods in Engineering*, 74:1897–1927, 2008.
- [9] X. Chengxian and J.L. Jong. Sequential quadratic programming methods for optimal control problems with state constraints. *Applied Mathematics - A Journal of Chinese Universities*, 8(2):163–174, 1993.
- [10] J. Chung and G.M. Hulbert. A time integration algorithm for structural dynamics with improved numerical dissipation: The generalized- $\alpha$  method. *ASME Journal of Applied Mechanics*, 60:371–375, 1993.
- [11] S. Devasia, D. Chen, and B. Paden. Nonlinear inversion-based output tracking. *IEEE Transactions on Automatic Control*, 41(7):930–942, 1996.
- [12] M. Diehl, H.G. Bock, H. Diedam, and P.-B. Wieber. Fast direct multiple shooting algorithms for optimal robot control. In *Fast Motions in Biomechanics and Robotics*. Springer Berlin / Heidelberg, 2007.
- [13] M. Géradin and A. Cardona. *Flexible Multibody Dynamics: A Finite Element Approach*. John Wiley & Sons, New York, USA, 2001.
- [14] M. Hiller and A. Kecskemethy. Dynamics of multibody systems with minimal coordinates. In M.F.O. Pereira and NATO ASI Series J.C. Ambrosio, editors, *Computer-Aided Analysis of Rigid and Flexible Mechanical Systems*, Troia, Portugal, June 27 - July 9 1994. Kluwer Academic Publishers.
- [15] A. Isidori. *Nonlinear Control Systems*. John Wiley & Sons, London, UK, 3rd edition, 1995.
- [16] D.-S Kwon and W. J. Book. A time-domain inverse dynamics tracking control of a single-link flexible manipulator. *Journal of Dynamic Systems, Measurement, and Control*, 116:193–200, 1994.
- [17] N.M. Newmark. A method of computation for structural dynamics. *Journal of the engineering mechanics Division (ASCE)*, 85:67–94, 1959.
- [18] S. Sastry. *Nonlinear Systems: Analysis, Stability and Control*. Springer, 1999.
- [19] R. Seifried. Integrated mechanical and control design of underactuated multibody systems. *Nonlinear Dynamics*, 67:1539–1557, 2012.
- [20] R. Seifried. Two approaches for feedforward control and optimal design of underactuated multibody systems. *Multibody System Dynamics*, 27(1), 2012.
- [21] R. Seifried, G.J. Bastos, and O. Brüls. Computation of bounded feed-forward control for underactuated multibody systems using nonlinear optimization. *Proceedings in Applied Mathematics and Mechanics (PAMM)*, pages 69–70, 2011.
- [22] R. Seifried and P. Eberhard. Design of feed-forward control for underactuated multibody systems with kinematic redundancy. In Ginzinger L. Ulbrich, H., editor, *Motion and Vibration Control: Selected Papers from MOVIC 2008*. Springer, 2009.
- [23] R. Seifried, A. Held, and F. Dietmann. Analysis of feed-forward control design approaches for flexible multibody systems. *Journal of System Design and Dynamics*, 5(3), 2011.

- [24] M. Spong, S. Hutchinson, and M. Vidyasagar. *Robot Modeling and Control*. John Wiley & Sons, 2006.
- [25] D. Taylor and S. Li. Stable inversion of continuous-time nonlinear systems by finite-difference methods. *IEEE Transactions on Automatic Control*, 47(3):537–542, 2002.
- [26] P. Wenger and D. Chablat. Kinematic analysis of a class of analytic planar 3-RPR parallel manipulators. *Computational Kinematics: Proceedings of the 5th International Workshop on Computational Kinematics*, pages 43–50, 2009.