University of Liège
Faculty of applied sciences
Department of Aerospace and Mechanical Engineering
Multibody and Mechatronic Systems Laboratory

Geometric modeling and inverse dynamics of flexible manipulators

Thesis submitted in fulfilment of the requirements for the degree of
Doctor in Engineering Sciences

by

Arthur Lismonde
Research fellow of the F.R.I.A
Research Engineer

June 2020

Members of the examination committee

Prof. Olivier Brüls – *Advisor*
University of Liège (Belgium)

Prof. Pierre Duysinx – *President of the committee*
University of Liège (Belgium)

Prof. Bernard Boigelot
University of Liège (Belgium)

Prof. Christophe Collette
University of Liège (Belgium)

Prof. Paul Fisette
Université catholique de Louvain (Belgium)

Prof. Robert Seifried
Hamburg University of Technology (Germany)

Abstract

Efficiency is a central question in robotic developments nowadays. Robotic manipulators are designed to be more energy efficient and safer for their surroundings. This leads to the integration of lightweight and flexible components in the manipulators. The resulting elastic behavior affects the overall precision of the system and needs to be considered in the design of control laws. This work aims at developing tools able to model the complex behavior of flexible spatial manipulators and generate appropriate controls to achieve a given task.

Flexible robotic manipulators are modeled using a finite element approach allowing the description of spatial systems in a general and systematic way. Differential geometry tools are used to describe the kinematics and dynamics of the flexible system: thanks to a formulation on the $SE(3)$ matrix group, also known as the group of rigid body motions, the equations describing the complex spatial motion of the system are expressed in the local frame of the system and show reduced non-linearities compared to traditional methods.

For trajectory tracking of robotic manipulators, the definition of appropriate control inputs to drive the system is often based on the inverse dynamics of the system. However, the inverse dynamics of flexible systems can exhibit unstable internal dynamic behavior that needs to be taken care of. Here, based on a model of the system, the inverse dynamics problem is solved by formulating it as an optimization problem: an objective function is minimized under constraints imposed by the tracking task. The proposed method is general enough to deal with systems having serial or parallel kinematic topologies, flexible joints and/or links and even systems with unstable internal dynamics.

The generated feedforward inputs are first applied on simulation examples. The resulting behaviors of the flexible systems are discussed and compared to examples available in the literature. Two real robotic system are then considered. It is shown that realistic inputs can be generated with the method and that a reduction of unwanted vibrations inside the arm is obtained. Also, the real output error from the prescribed trajectory is reduced.

Acknowledgement

This work is the result of several years of research during which I had the support of many people.

First, I would like to thank Olivier Brüls and all the colleagues of the Multibody and Mechatronics Systems Laboratory (past and current) of the University of Liège. Since I started to work in 2014, the number of team members and equipment in the laboratory has grown. Yet, Olivier has always managed to devote time to discuss and give judicious advice about my research. In parallel, the numerous talks with the colleagues allowed me to complete my thoughts, which led to the present work.

I also thank the members of the Institute of Robotics from the Johannes Kepler University Linz and the Institute of Mechanics and Ocean Engineering from the Hamburg University of Technology. Thanks to the discussions during my visit, I was able to enlarge my understandings on control and modeling of robots.

At last, I would like to thank Mélanie, my family and my friends who have always supported me these past years.

This work has been partially supported by the Belgian Fund for Research training in Industry and Agriculture (FRIA grant) and the “Robotix-Academy” project within the European INTERREG V A Greater Region program.

Contents

Chapter 1

Introduction

The application fields of robotics and automation have been growing fast these past years. Early robotic manipulators were constructed with very stiff and bulky links, requiring a separation between the robot's workspace and other human operators workspace. Today, the use of automated manipulators is no longer limited to such controlled environment. In fact, robotic manipulators are now integrated in environments where human collaborators are present. Such human-robot interaction applications require to improve both safety and efficiency of robotic arms. For example, the Sawyer robot is a *collaborative robot* meant to share its workspace with other human operators in a production process (see Fig. 1.1). In medical applications, the Da Vinci robot (see Fig. 1.2) is a surgical robot that assist the surgeon and the medical staff to perform minimally invasive surgeries. Robots are also used in daily life to assist people. The JACO robotic arm (Fig. 1.3) assists people with limited upper limb mobility. With such considerations, today's robotic manufacturers tend to use lightweight structures and increase the compliance of robotic systems. As a result, robotic manipulators can become intrinsically safe as the severity of potential collisions is reduced.

Robotic manipulators are usually made up of members, called links, connected by articulated joints. Each joint allows certain relative motions between the connected links. The system is thus naturally modeled as a multibody system (MBS). By actuating the joints of the system, one is able to drive the end-effector of the robot to desired targets. Attaching a tool to this end-effector allows the complete robotic system to perform various tasks. Analyzing the environment of the task is essential to plan efficient motions required to achieve it: this is called the path planning process. Usually, two types of paths can be distinguished: a point-to-point motion (PTP) or a trajectory tracking motion. In PTP motions, only the final

Figure 1.1: Sawyer, a collaborative robot from Rethink Robotics, Hahn group (picture from http://www.made-in-europe.nu/2017/05/eerst-sawyer-cobot-met-ce-goedkeuring-in-de-benelux/).

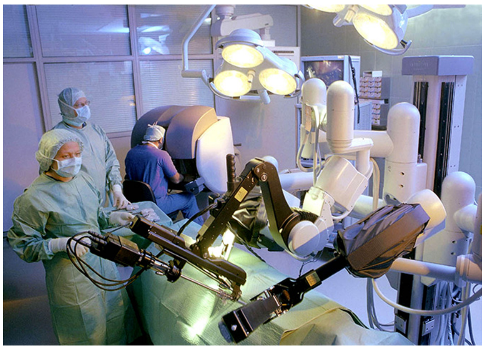


Figure 1.2: Da Vinci surgical robot from Intuitive Surgical (picture from www.wired.com/2009/09/surgical-robots/2/).

Figure 1.3: Kinova JACO® assistive robot from Kinova (picture from www.kinovarobotics.com).

configuration of the manipulator is of importance and the transient behavior is not much of interest. In this case, one often looks for the fastest or the most economical path between two configurations of the system. For trajectory tracking, on the other hand, the configuration of the manipulator is specified at any time along the trajectory and, therefore, the transient behavior of the manipulator between the initial time and final time is important. Regardless of the type of motion that is chosen, there are two main ways to specify it. As explained in [84], the motion of the system can either be specified in the joint space or in the task space [31]. Since robotic manipulators are usually controlled through their joints, specifying the motion in the joint space seems quite natural: the instructions are given in terms of the position of each joints. On the other hand, in most robotic applications, the desired task requires the positioning of a tool in the task space. Therefore, the latter is actually more convenient when planning a path. Once the desired task and motion are defined, one can look into the design of the control problem; whose purpose is to provide suitable inputs to the system to drive it along a desired motion despite outside disturbances. The whole process of a general robotic task is shown in the block diagram of Fig. 1.4. Generally speaking, the controller can also be affected by the robot behavior. As a consequence, it is often required to understand the dynamics between the actuated joints and the end-effector. This is even more critical for lightweight and compliant robots with elastic links and joints. Although such robots can exhibit attractive features, their elasticity can negatively impact the precision and stability of the system. As a result, the dynamics of the system can be more challenging to control.

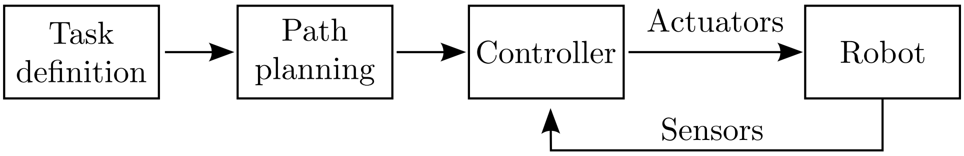


Figure 1.4: Block diagram of a robotic task.

This thesis presents a contribution to the control design of flexible manipulators for trajectory tracking tasks. Relying on a geometric formulation of the dynamics of flexible systems, efficient control laws are generated by solving their inverse dynamics. In the coming sections, various control strategies for flexible manipulator are first presented. Then, modeling techniques useful for the description of the system's dynamics are summarized. Next, some optimization solvers are presented. Eventually, the objective and contributions of the present work are given.

1.1 Control design of mechanical systems

In general, a mechanical system has some actuators, or inputs, that induce a motion at the outputs of the system to achieve a given task. In mechanical systems, flexibility can be present locally, e.g. when a spring is used to connect two rigid components, or distributed in a structural component, e.g. when a slender and deformable link is used. The motion of a mechanical system can be described by rigid modes and flexible modes. If a system only has rigid modes, the mechanical systems is a rigid system. Depending on the number of *degrees of freedom* (dof) and actuators of the system, a rigid system can be fully actuated, under-actuated [14, 15, 99] or over-actuated [48, 49]. In fully actuated rigid systems, the actuators directly affect the rigid modes of the system. If a system only has flexible modes, it can be considered as a flexible structure. The elastic motion can be controlled by placing a number of actuators on the structure [30, 94]. When the actuators and the outputs are located at the same place, one talks about *collocated* systems, otherwise, one talks about *non-collocated* systems. In the present work, flexible systems which have both rigid and flexible modes are considered. It is assumed that the number of actuators is equal to the number of rigid modes in the system and that the flexible modes are only indirectly controlled by the actuators. It is also assumed that the number of outputs is equal to the number of actuators but that the actuators and outputs are non-collocated. These assumptions are valid for a wide class of robotic manipulators such as the Sawyer, Da Vinci or Jaco robots shown previously. As a result, the flexible systems considered here have more dofs than actuators and are under-actuated systems.

An example of a robot manipulator controller is depicted in Fig. 1.5. The prescribed motion $\mathbf{y}_{\text{presc}}$ of the output $\mathbf{y}$ originates from the path planning process (see Fig. 1.4). Based on a model, the feedforward control computes the reference actuator positions $\mathbf{q}_{\text{M,FF}}$ and reference actuator efforts $\mathbf{u}_{\text{FF}}$. The feedback control effort $\mathbf{u}_{\text{FB}}$ corrects the actual positions of the robot based on a measure of the error. Therefore, the control efforts $\mathbf{u} = \mathbf{u}_{\text{FF}} + \mathbf{u}_{\text{FB}}$ sent to the robot originate from two components: a feedforward control $\mathbf{u}_{\text{FF}}$ and a feedback control $\mathbf{u}_{\text{FB}}$.

In the coming sections, a few techniques used for the design of feedback control laws are first reviewed. Then feedforward design techniques for systems with various characteristics are presented.

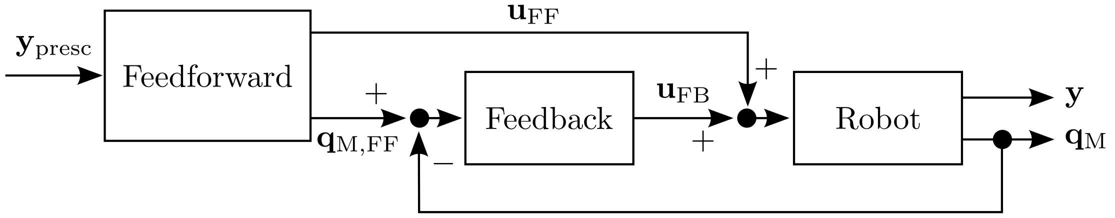


Figure 1.5: Schematic view of a general controller.

1.1.1 Feedback control

Unpredicted disturbances from the environment can easily affect the control of a mechanical system. In order to have a desired output behavior, such disturbances can be compensated using a feedback controller. The output and state errors of the arm can be monitored using sensors or estimators [86]. Appropriate compensation $\mathbf{u}_{\mathrm{FB}}$ can then be designed.

In [28], a feedback control law is constructed based on a measurement of the end-effector position using an optical device. Additional care was taken to reduce noise from ambient light conditions. To monitor the elastic deflections of the flexible links, strain gauges [46, 72, 80] or optical systems [60] can be used. In [111], accelerometers fixed at the end-effector of the ELLA robot are used to estimate the states of the system required for the feedback control.

Once the error is measured, classical proportional-integral-derivative (PID) feedback and its variants can be implemented quite easily on each joint independently. Linear quadratic regulators (LQR), such as in [56], are also widely used in rigid and flexible manipulators. In [92], the combination of a Kalman filter and a sliding mode technique is used to estimate and control the states of a flexible manipulator in a robust way. A sliding mode strategy has also been used in [69] for the control of a two link flexible manipulator. Feedback linearization, which linearizes the states of a non-linear system thanks to a change of variable, have been applied in [110] for the control of elastic robots.

Eventually, all feedback techniques react to existing error appearing in the system in order to insure a robust control of the output. To complement this action, a feedforward control can be designed to generate motions and control inputs that have reduced error in the open loop and undisturbed case.

1.1.2 Feedforward control

In parallel to the feedback control, the feedforward control generates the reference actuator position $\mathbf{q}_{\mathrm{M,FF}}$ and reference actuator efforts $\mathbf{u}_{\mathrm{FF}}$. Without outside disturbances and errors, these reference inputs are constructed to obtain an ideal behavior of the system.

By analyzing the dynamics of a flexible system, the inputs can be processed with filtering techniques to avoid exciting the resonance frequencies of the system. For example, using low-pass or band-stop filters with appropriate frequency parameters [81], no energy corresponding to resonance frequencies is transmitted and little residual vibrations are induced in the system. Such filtering technique requires little computation time but may limit the actuation bandwidth, which penalizes the performance of the control system.

Input shaping methods, explained in [104, 80, 34, 30], require the analysis of system's impulse response. With appropriate amplitudes and delays [71], a sequence of impulses, called the input shaper, is constructed. A new input is then generated by convolving any desired input with the input shaper, leading to a sum of scaled and delayed desired inputs. This resulting shaped input enable to reduce the residual vibrations in the system at the end of the trajectory as it is shown for the TUDOR robot in [72].

The above filtering and shaping techniques are mainly designed for linear systems. Also, they are mostly applied for PTP motions and less adapted for trajectory tracking problems. In the latter, the transient behavior of the system during the motion is important. Therefore, in trajectory tracking problems, the definition of the control inputs can be stated as an inverse dynamics problem, defined as follows:

Given a prescribed output trajectory $\mathbf{y}_{\mathrm{presc}}(t)$*, what are the inputs* $\mathbf{u}_{\mathrm{FF}}(t)$ *that will drive the actual output* $\mathbf{y}(t)$ *along this desired trajectory?*

A brief overview of techniques used to solve such inverse dynamics problem is now given. A more comprehensive explanation of the applicability of each technique will be detailed in Chapter 2.

For the case of fully actuated rigid manipulators, one well spread method is the *computed torque method* [31, 47]. This method directly substitutes the output $\mathbf{y}$ in the equations of motion of the system by the prescribed trajectory $\mathbf{y}_{\mathrm{presc}}(t)$. Since the system is fully actuated, the input efforts $\mathbf{u}_{\mathrm{FF}}$ are directly given by the equations of motion and correspond to a compensation of the inertia, gyroscopic and external forces applied on the system.

Some under-actuated manipulators are classified as flat systems, so that flatness based methods can be used. A system is flat if one can find a flat output of the system for which the prescribed trajectory $\mathbf{y}_{\mathrm{presc}}(t)$ defines the control inputs and all states of the system in an algebraic way [34, 111]. Flatness based methods are strongly related to feedback linearization methods as it is discussed in [116, 117]. However, finding such flat output may not be straightforward or even possible in general.

Many under-actuated systems are not differentially flat and, in such cases, the input is influenced by the internal dynamics of the system as it will be detailed in Chapter 2. The stability of the internal dynamics strongly affects the computation of the inverse dynamics solution. This stability can depend on several factors: the localized or distributed nature of the flexibility, the parameters of the system, the localization of the actuators, the choice of the system's output, etc. If the internal dynamics of the system is stable, the system is minimum phase and the inputs $\mathbf{u}_{\mathrm{FF}}$ of the system can be found by time integrating the equations of motion from an initial state [58, 78, 98].

In contrast, if the internal dynamics of a system is unstable, the system is non-minimum phase and it can lead to unbounded solutions if not dealt with properly. Here is a review of some techniques able to deal with non-minimum phase systems.

Instead of dealing directly with the non-minimum phase characteristic of the system, an alternative output can be chosen so that the resulting system becomes flat or at least has minimum phase behavior [35, 119]. In [77, 98] an alternative output which is close to the actual output is proposed.

For linear and non-minimum phase systems, the time domain inverse dynamics method [65] separates the dynamics of the system into a causal and an anti-causal part before the inverse dynamics is solved. The causal part leads to some post-actuation in the resulting inputs while the anti-causal part leads to some pre-actuation in the system. In [9, 83], the inverse dynamics of a single link non-minimum phase system is solved by defining the dynamics in the frequency domain. While the time domain and the frequency domain inverse dynamics methods lead to non-causal solution, a method presented in [11] finds a causal solution of the inverse dynamics of non-minimum phase and linear systems. This method relies on defining the solution as a polynomial function with linear coefficients. Some conditions on these coefficients force the unstable behavior of the system to vanish.

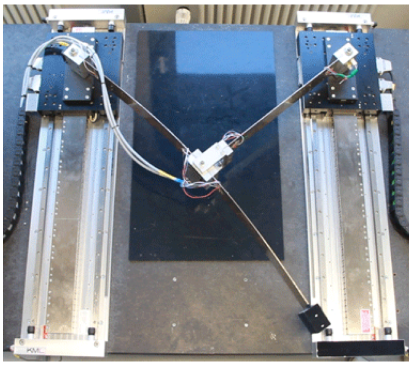


Figure 1.6: Lambda kinematics parallel robot with 2 flexible links from the University of Stuttgart, Germany [82, 98].

Alternatively, so called stable inversion techniques can be used to directly compute the inputs of non-minimum phase and non-linear system [29, 40]. In [53, 100, 112], the inverse dynamics problem is stated as a boundary value problem (BVP). The definition of the boundary conditions is based on the analysis of the stable and unstable manifolds of the system. The former is defined as a portion of the state space where all trajectories converge to a given configuration as time goes forward. The latter is a portion of the state space where all trajectories converge to a given configuration as time goes backward. Then, the initial condition of the BVP forces the system to start on an unstable manifold and the final condition assures that the system ends on a stable manifold. The resulting solution of the inverse dynamics problem is bounded but non-causal. Indeed, the unstable and stable characteristics respectively lead to some actuation of the system before (pre-actuation) and after (post-actuation) the actual tracking of the trajectory. This formulation is successfully applied [82, 98] on a highly flexible planar system depicted in Fig. 1.6. However, it can be noted that the construction of such manifolds is based on the eigen properties of the linearized system and can be quite cumbersome to compute. As stated in references [22, 62, 98], the stable inversion technique requires that the equilibrium point around which the linearization is performed is hyperbolic, i.e. the linearization has no eigenvalues on the imaginary axis. This insures that the stable and unstable manifolds can be determined [105] and that the durations of the pre- and post-actuation phases are finite. If the internal dynamics is not hyperbolic, it is possible to modify the internal dynamics to have a hyperbolic behavior [39].

In order to avoid the definition of boundary values, the inverse dynamics problem can be stated as an optimization problem [5, 6]. The problem is then formulated as the minimization of a cost function under the constraints that the equations of motion of the system and tracking conditions are satisfied. Such cost function represents a measure of the amplitude of the internal dynamics. The solution of this optimization formulation also presents non-causal characteristics with some pre- and post-actuation of the system. It was shown in [7] that the optimization formulation converges to the BVP formulation as the length of the pre- and post-actuation phases increases. Optimization strategies were also chosen in [79, 59] to solve the dynamics of flexible robot manipulators.

In the present work, an optimization based formulation is chosen to solve the inverse dynamics problem of general flexible manipulators. Indeed, compared to a BVP formulation, such optimization formulation benefits from a more general and simpler definition of the problem.

1.1.3 Experimental control

Extensive reviews of the analysis of the dynamics and control of flexible robots can be found in [12, 33, 42, 68, 91, 96]. Experiments on the control of flexible robots are usually performed on planar 2D systems [2, 26, 28, 30, 65, 74, 87]. To the best of our knowledge, experiments on flexible 3D robots such as in [46, 72, 111] is less developed.

Since such spatial manipulators are more practical for real life and industrial applications, the present thesis aims at contributing to the experimental control of spatial flexible manipulators.

1.2 Models for flexible MBS

In order to apply the control methods described previously, it is necessary to describe the dynamics of the systems mathematically including flexibility effects. In engineering problems, one often relies on the construction of a model describing the kinematics and the dynamics of the system at hand. For MBS, such models involve kinematic joints connecting the different bodies. Each body is usually described by a set of coordinates $\mathbf{q}$. The interactions of each body with its surrounding is modeled through forces which can be externally applied loads, e.g. gravity, internal connecting forces within bodies or additional forces generated by other elements connected to the bodies. More particularly, some actuating forces $\mathbf{u}$ may be present to drive the system to achieve a given task. Various techniques are possible to model flexibility effects in MBS, such as: localized flexibility inside a joint, so called flexible joint models, and distributed flexibility along a link, so called flexible link models. A review of the main methods is presented below.

1.2.1 Flexible joint models

Flexibility inside joints is usually the result of elasticity in the transmission chain between the actuator and the joint [52, 85], e.g. shafts torsion, gear wheel deflection, belt elasticity. To model such flexibility, additional dofs should be defined in the joint. The robot links are usually considered as rigid bodies. An example of such flexible joint is shown for the case of a hinge joint in Fig. 1.7: an elastic component links the motor relative coordinate $\mathbf{q}_M$ to the relative deflection of the transmission between the motor and the link $\mathbf{q}_L$. Additionally, imperfections and non-linear effects can be represented too [52]. These effects can include non-linear elastic behavior and frictional effects that can be represented by viscous, Coulomb or Stribeck models [41, 118]. Reference [110] studies such models for a single link example. For multi-link and spatial examples, an extended flexible joint model was proposed in [78, 79] and takes into account elastic deformations in non-driven direction as well.

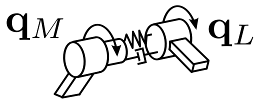


Figure 1.7: Flexible revolute joint model with a motor angle $\mathbf{q}_M$ and a link angle $\mathbf{q}_L$ linked through a torsional spring-damper pair.

1.2.2 Flexible link models

Aside from the joint elasticity, the links of the robots are usually constructed with materials having a finite stiffness. Depending on the material and the geometry of the links, the flexible behavior of the link can be more or less predominating. It is therefore important to be able to include such behavior in the model.

The lumped element method (LEM)

Link flexibility can be modeled by concentrating it in an equivalent spring damper element located at its connecting joint. The stiffness and damping parameters of these local elements are calibrated based on the distributed parameters of the flexible link [111]. A more general approach is to model a link as a succession of rigid bodies and flexible non-actuated joints as shown in [36, 46, 87].

The assumed mode method (AMM)

According to the AMM, the motion of the system is described by two types of coordinates: rigid body coordinates describing the overall rigid body motion and elastic coordinates describing the deflection around this rigid motion. The global elastic deflection of the arm is thus approximated using a finite series of shape functions spanning the entire link, e.g., the natural modes shapes of the system. Each elastic coordinate represent the amplitude of the considered mode shape. In [11, 28, 35, 65, 90, 106], examples of robots with a single flexible link modeled using the AMM are shown. Planar and multi-link cases were studied in [3, 37, 56, 74] for open chain kinematics and in [26] for closed chain kinematics. The extension of this method to 3D manipulators is used in [50, 57]. A comparison of this method with the FEM method, presented hereafter, can be found in [113]. In [16, 17], such AMM is used in combination with 4×4 transformation matrices, as first developed for rigid systems in [38], to describe the kinematics of the manipulator.

The finite element method (FEM)

As an alternative to the widespread AMM, the FEM can also be used to model flexible links. In this method, which is widely used to model flexible MBS, the link is discretized in space into elements and nodes, described by nodal coordinates. Local interpolation functions then define the deflection between the nodes. Here, one possibility is to describe the flexible behavior of the element as a flexible motion around an overall rigid motion according to the floating frame of reference approach [9, 10, 59, 66, 97, 102]. An alternative method, the so-called geometrically exact formulation, does not rely on any separation between the rigid and the flexible motion [8, 51, 103]. The coordinates are directly described in a global inertial frame and no distinction is made between rigid or flexible variables. Compared to the floating frame of reference approaches, the non-linearities in the inertia forces are reduced at the expense of higher non-linearities in the internal forces of the dynamics equations [24, 120]. It can be noted that the FEM usually leads to a higher dimensional system compared to AMM. However, a FEM model has a sparse structure and allows a more systematic approach to deal with multi-link systems with closed chain kinematics and non-linear deflections.

Recently, in order to reduce non-linearities in the internal forces of geometrically exact finite element models, a local frame approach has been proposed [21, 107, 108]. Relying on a description of the deformation and velocity fields in the local frame of each body, this approach benefits from several advantages. The inertia forces, internal forces and kinematic constraints of the system are naturally invariant under superimposed Euclidean transformations. As a result, the non-linearities due to large amplitude motions are reduced. From a numerical point of view, this also induces important reductions of the computational complexity of the problem. Thanks to a global parameterization-free formulation of the equations, singularity issues arising from the description of the rotations are avoided. It is also shown in [109] that flexible elements do not suffer from shear locking phenomenon. The kinematics of the system is represented in terms of 4×4 transformation matrices which belong to a Lie group. These models are considered as geometric models and are developed using differential geometry concepts. This local frame FEM approach has been applied in the forward dynamic simulations of flexible robots in [55, 54] but has not yet been applied to control problems.

Some combinations of flexible joint and flexible link model are also reported in the literature. The AMM is used in [89] to simulate a single flexible link and flexible joint manipulator. In [43], a FEM approach is chosen to model spatial manipulators with multiple flexible links and flexible joints.

1.2.3 Models for inverse dynamics

Eventually, each modeling approach is able to grasp the characteristics of a real robot with a variable level of detail and generality.

In the flexible joint case, if no damping is considered, the model can be classified as a flat system. On the other hand, flexible link models have usually some internal dynamics whose stability properties can strongly affect the model inversion procedure (see Chapter 2).

In order to represent realistic manipulators that may include distributed and localized flexibility, a modeling method able to consider both types of flexibility needs to be considered. In the present work, the previously defined geometric and local frame FEM models are adopted and extended to control problems. Indeed, the generality and numerical benefits of such approach allow one to state efficiently the inverse dynamics problem of flexible systems.

1.3 Optimization solvers

In addition to the formulation of control problems and modeling techniques, the question on how to solve the resulting trajectory tracking problem arises. Here, as the inverse dynamics problem is stated as an optimization problem, a brief overview of optimization algorithms is given. Optimization algorithms aim at minimizing a given objective function $J(\mathbf{x}, t)$ which depends on the system's design variables $\mathbf{x}(t)$ (the states $\mathbf{q}(t)$ and the inputs $\mathbf{u}(t)$). In parallel, it has to satisfy some constraints defined by the dynamics of the system and the tracking conditions.

Two types of methods are usually considered [13, 63]: direct methods, which use a *discretize then optimize* philosophy, and indirect methods, which use an *optimize then discretize* philosophy. In direct methods, all states and inputs of the system are first discretized in time, or *transcribed*, leading to a so-called *non-linear program* (NLP): a constrained optimization problem with a non-linear objective function and/or constraints. The algorithm then iteratively generates optimized states and inputs until a given tolerance is satisfied. In indirect methods, analytical optimality conditions are first computed before discretizing and solving the problem. While the latter usually leads to a more accurate solution, direct methods are more robust and easier to implement. Amongst direct methods, one can further distinguish direct shooting methods [19, 22] from direct transcription methods [5, 18]. Shooting methods relies on dynamic simulations over discrete time intervals, also called shooting intervals. Smaller time steps can be used for the simulations within each shooting interval. As a result, a finer time grid for the final solution is obtained. Shooting methods, however, are more sensitive to the numerical parameters and poor initial guesses. Direct transcription methods, on the other hand, directly deal with the state variables at the discrete time steps. This generally leads to a problem with a rather large dimension but a sparse structure. Compared to shooting methods, the convergence of direct transcription methods is more robust and therefore more suitable to solve general non-linear dynamic problems.

Traditional optimization solvers [13] have their design variables defined on a linear vector space. However, spatial geometric models have variables defined on a non-linear Lie group (see Section 1.2). Optimization solvers defined on non-linear manifolds are described in [1, 73]. In the present work, the extension of such non-linear optimization solver to inverse dynamics problem and geometric models is presented.

1.4 Objective and contributions of the thesis

This work aims at developing a method able to model general flexible 3D manipulators and solve the inverse dynamics problem. It further aims at an experimental validation of the method with flexible spatial robots.

The contributions can be summarized as follows:

- It is first shown how the geometric FEM, initially presented in [21, 108, 107], can be applied to the modeling of robotic systems. Thanks to the generality of the FEM approach, models of spatial robots with various configurations (flexible links and/or joints, open or closed chain kinematics) are constructed in a systematic way.

- Secondly, a geometric approach is proposed to formulate the inverse dynamics problem as a constrained optimization problem. This approach extends the method initially presented in [4, 5, 6] for planar problems. By taking advantage of the local frame representation, the equations of motion have reduced non-linearities which positively impacts the formulation of the feedforward control problem of 3D systems. Relying on differential geometry tools, the non-linear design variables of the optimization problem are redefined on the Lie algebra so that classical optimization tools can be used.

- Then, the impact of the internal dynamics of the system on the inverse dynamics solution is investigated for simple linear examples as well as more complex spatial examples.
- Eventually, the present methodology is exploited to design and implement feedforward controllers in real spatial applications. The performances of the present method are analyzed for two experimental robotic manipulators.

1.5 Outline of the thesis

The structure of the thesis is as follows:

- Chapter 2: Fundamental concepts regarding the inverse dynamics problem and the internal dynamics of a mechanical system are discussed. Several linear examples are used to illustrate these concepts.
- Chapter 3: Some theoretical mathematical tools required to understand the model formulation are first given. The formulation of the equations of motion of a MBS in the $SE(3)$ group is presented. Then, the construction of the FEM model with such formulation is shown. An algorithm to solve the direct dynamic problem in the $SE(3)$ group is also detailed.
- Chapter 4: The inverse dynamics problem is presented for systems with rigid and flexible components. The geometric formulation of the optimization problem is developed. Then the algorithm and necessary derivations to compute the solution are shown. To illustrate the method, a planar under-actuated cart system [100] is considered.
- Chapter 5: The method is applied and analyzed for two spatial flexible manipulators: a serial and a parallel manipulator. The influence of the flexible characteristics and internal dynamics on the solution of the inverse dynamics problem is discussed.
- Chapter 6: After the simulation studies, the inverse dynamics of real flexible manipulators is investigated. First, a flexible link robot, ELLA, is tested experimentally and the vibrations in the robot arm are analyzed for different reference inputs. Then, a flexible joint robot, Sawyer, is considered. The end-effector tracking precision is compared between a rigid and a flexible reference input trajectory.

- Chapter 7: The conclusions on the present methodology are drawn. Some perspectives on the further developments of the present inverse dynamics solver and feedforward control strategy are also presented.

Chapter 2

Inverse dynamics of simple mechanical systems

The aim of this chapter is to present some fundamental concepts about the control of mechanical systems. First, the general equations of the dynamics of mechanical system is recalled. Then, the concepts of direct and inverse dynamics problems are stated. Lastly, the concept of internal dynamics is illustrated for a few illustrative examples.

2.1 Dynamics of mechanical systems

The motion of each part of a mechanical system is often described by a vector of coordinates $\mathbf{q}$. Under the action of some input efforts $\mathbf{u}$, the system evolves over time and result in some outputs $\mathbf{y}(\mathbf{q})$ that can be a function of the coordinates. The general equations governing the dynamics of such system is defined as

$$\mathbf{M}\ddot{\mathbf{q}} + \mathbf{g}(\mathbf{q}, \dot{\mathbf{q}}, t) = \mathbf{A}\mathbf{u}(t) \tag{2.1}$$

$$\mathbf{y} = \mathbf{h}(\mathbf{q}(t)) \tag{2.2}$$

where the $\mathbf{M}$ matrix defines the inertia of the system, the vector $\mathbf{g}$ represents the internal and external forces applied on the system, the matrix $\mathbf{A}$ applies the inputs $\mathbf{u}$ on the appropriate coordinates and the function $\mathbf{h}(\mathbf{q})$ defines the outputs $\mathbf{y}$ of the system. From here, one can now express the direct and inverse dynamics problems.

2.1.1 Direct dynamics problem

In a direct dynamics problem, the inputs are prescribed, i.e. $\mathbf{u} = \mathbf{u}_{\text{presc}}(t)$, and the behavior of the outputs $\mathbf{y}$ is unknown; *Given a prescribed input* $\mathbf{u}_{\text{presc}}(t)$*, what is the behavior of the outputs* $\mathbf{y}(\mathbf{q}(t))$ *of the system?*

The previous equations thus become

$$\begin{aligned} \mathbf{M}\ddot{\mathbf{q}} + \mathbf{g}(\mathbf{q}, \dot{\mathbf{q}}, t) &= \mathbf{A}\mathbf{u}_{\text{presc}}(t) \\ \mathbf{y} &= \mathbf{h}(\mathbf{q}(t)) \end{aligned} \tag{2.3}$$

In order to find a solution for the output $\mathbf{y}$, the dynamics of Eqs. (2.3) can be solved starting from some given initial states of the system.

2.1.2 Inverse dynamics problem

Conversely, in an inverse dynamics problem, the inputs $\mathbf{u}$ are unknown and the outputs are prescribed, i.e. $\mathbf{y} = \mathbf{y}_{\text{presc}}(t)$; *Given a prescribed output trajectory* $\mathbf{y}_{\text{presc}}(t)$*, what are the inputs* $\mathbf{u}(t)$ *that will drive the actual output* $\mathbf{y}(\mathbf{q}(t))$ *along this desired trajectory?*

The equations become

$$\mathbf{M}\ddot{\mathbf{q}} + \mathbf{g}(\mathbf{q}, \dot{\mathbf{q}}, t) = \mathbf{A}\mathbf{u}(t) \tag{2.4}$$

$$\mathbf{y}_{\text{presc}}(t) = \mathbf{h}(\mathbf{q}(t)) \tag{2.5}$$

Here, the actual output is a function of the coordinates of the system which have to satisfy the equations of motion, i.e. $\mathbf{y} = \mathbf{h}(\mathbf{q}(t))$. Mathematically, the output tracking problem is defined by Eq. (2.5), called the *servo-constraints* [14, 15, 99] and reformulated as:

$$\mathbf{y} - \mathbf{y}_{\text{presc}}(t) = \mathbf{0} \tag{2.6}$$

In this case, Eqs. (2.4-2.5) are the equations of the internal dynamics of the system and a solution for the inputs $\mathbf{u}(t)$ must be found.

2.1.3 DAE index and relative degree

Considering that the outputs $\mathbf{y}$ do not depend on higher order derivatives of the coordinates $\mathbf{q}$, one can note that the above equations are a set of differential algebraic equations (DAE). The differential index of such DAE is 3 [27], i.e. in order to have ordinary differential equations (ODE), one needs to differentiate Eq. (2.6) three times. This index is representative of the difficulty to solve the system numerically: the higher the index, the more the solution is prone to numerical instabilities. In order to evaluate the index of the DAE representing the internal dynamics, one can construct the following matrix

$$\begin{bmatrix} \mathbf{M} & -\mathbf{A} \\ \mathbf{D} & \mathbf{0} \end{bmatrix} \tag{2.7}$$

where $\mathbf{D}$ is the gradient of Eq. (2.6) with respect to the coordinates $\mathbf{q}$. If this matrix is non-singular, the internal dynamics is represented by an index 3 DAE. If this matrix is singular, the index is higher than 3. Some techniques [14, 121] can be used to reduce the index of such DAE. Reducing the index by introducing constraints on velocity or acceleration level, leads to systems that are easier to solve numerically.

From a control point of view, an additional relevant concept is the relative degree of a system [27, 62]. The relative degree indicates how one of the inputs is related to one of the outputs and its derivatives. The relative degree depends on the configuration of the system and the chosen output [88]. One way of interpreting the relative degree is to consider it as a measure of how *direct* the input is affecting the output. The lower the relative degree the *faster* the input affects the output. This concept can be extended to the case of multiple inputs, multiple outputs (MIMO) systems [105]. One then defines the *total relative degree* as the sum of the relative degree of each input/output pair.

For all the flexible link systems considered in this work, the inverse dynamics is expressed as an index 3 DAE. However, their relative degree is not always easy to define as it will be discussed later.

2.2 Internal dynamics

Recall from the introduction that the flexible systems considered in this work are under-actuated systems with localized or distributed flexibility. A classification of such systems is depicted in Fig. 2.1. Amongst those systems, one can distinguish systems that are *differentially flat* and systems that have some *internal dynamics.* Differentially flat systems [44, 45] are defined as systems for which *all states and inputs can directly be expressed algebraically in terms of a so-called flat output and its derivatives.* The design of control inputs for such systems is greatly simplified since no dynamics needs to be integrated. On the other hand, the internal dynamics of a system is defined as *the remaining dynamics when the outputs of the system are* *prescribed (and possibly varying).* Such internal dynamics can result from the non-collocated nature of inputs and outputs [34, 36, 56, 77]. Indeed, unlike ideal rigid structures in which wave propagation velocity is infinite, i.e. a force is transmitted instantaneously from one end of the structure to the other, the wave propagation velocity in real flexible structure is finite. As a consequence, the more the structure is flexible, the higher the delay between the actuation of the arm and the end-effector motion. This leads to a phase lag in the response of the system which can destabilize the control actions. Depending on the importance of the phase lag, the dynamics inside the system can be *minimum phase*, i.e. the internal dynamics is stable, or *non-minimum phase*, i.e. the internal dynamics is unstable. Such characteristic can strongly affect the design of the controller of the system. A stability analysis of the non-linear internal dynamics is, in practice, limited to the zero dynamics of the linearized system [62, 95] which is defined as *the remaining dynamics when the outputs are fixed to a constant equilibrium point.* The linear analysis of the internal dynamics done in reference [22] is now summarized.

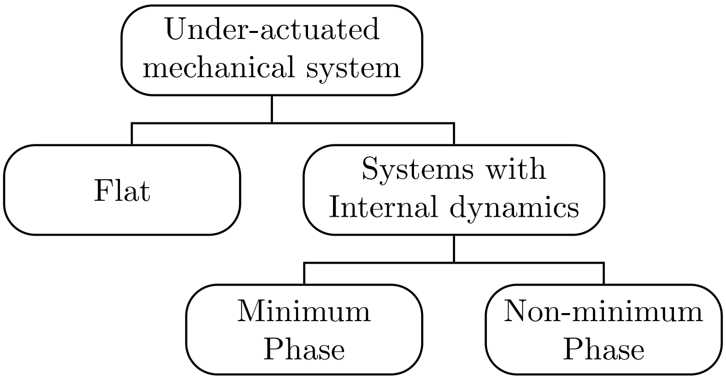


Figure 2.1: Classification of under-actuated mechanical systems.

2.2.1 Linear analysis

The non-linear system of Eqs. (2.4-2.5) can also be written in a descriptor state space form as

$$\mathbf{M}^{\star}\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, t) \qquad \text{with } \mathbf{x} = \begin{bmatrix} \mathbf{q} \\ \dot{\mathbf{q}} \\ \mathbf{u} \end{bmatrix} \tag{2.8}$$

where $\mathbf{x}$ gathers the states of the system, $\mathbf{M}^{\star}$ is a singular matrix and the information about the output is contained in $\mathbf{f}$. This system can be linearized around an equilibrium point with some increments $\Delta\mathbf{q}$, $\Delta\dot{\mathbf{q}}$, $\Delta\ddot{\mathbf{q}}$ and $\Delta\mathbf{u}$. The linearization leads to

$$\mathbf{M}^\star \begin{bmatrix} \Delta\dot{\mathbf{q}} \\ \Delta\ddot{\mathbf{q}} \\ \Delta\dot{\mathbf{u}} \end{bmatrix} = \mathbf{F}^\star \begin{bmatrix} \Delta\mathbf{q} \\ \Delta\dot{\mathbf{q}} \\ \Delta\mathbf{u} \end{bmatrix} \tag{2.9}$$

with

$$\mathbf{M}^\star = \begin{bmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{M} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} \quad ; \quad \mathbf{F}^\star = \begin{bmatrix} \mathbf{0} & \mathbf{I} & \mathbf{0} \\ -\mathbf{K} & -\mathbf{C} & \mathbf{A} \\ \mathbf{D} & \mathbf{0} & \mathbf{0} \end{bmatrix} \tag{2.10}$$

and the matrices $\mathbf{C} = \partial\mathbf{g}/\partial\dot{\mathbf{q}}$ and $\mathbf{K} = \partial\mathbf{g}/\partial\mathbf{q}$ are the linearization of the internal and external forces $\mathbf{g}$ applied on the system.

The stability of the zero dynamics of the linearized system can then be determined by an analysis of the generalized eigenvalues problem of Eq. (2.9) around given equilibrium points. If such equilibrium points are hyperbolic, i.e. the linearized system has no eigenvalues on the imaginary axis, they can be characterized as stable or unstable if the real part of the eigenvalue is negative or positive, respectively.

As already stated in the introduction, depending on the characteristics of the system, various techniques can be used to find such solution. A few examples are now given.

2.3 Illustrative examples

2.3.1 Fully actuated systems

Let us first consider the case of a fully actuated rigid system for which the computed torque method can be applied (see introduction). It is a simple cart system depicted in Fig. 2.2. It has a mass m, one dof q and one actuating input u. The output y of the system is its only dof, i.e. $y = q$.

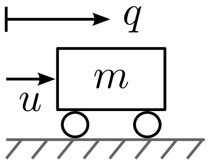


Figure 2.2: Fully actuated cart with one dof q and one control input u.

The inverse dynamics problem of the system is governed by dynamic equation of the system and the servo-constraints (Eq. (2.6)), leading to

$$m\ddot{q} = u \tag{2.11}$$

$$q = y_{\text{presc}}(t) \tag{2.12}$$

Assuming the prescribed trajectory $y_{\text{presc}}(t)$ is sufficiently smooth, the scalar input effort u_{FF} can directly be computed as

$$u_{\text{FF}}(t) = m\,\ddot{y}_{\text{presc}}(t) \tag{2.13}$$

and is the solution of the inverse dynamics problem.

The differential index of this fully actuated system can be determined by constructing the matrix given by Eq. (2.7). In this case, $\mathbf{M} = m$, $\mathbf{D} = 1$ and $\mathbf{A} = 1$ and the matrix is not singular and the index of the resulting DAE is 3. Also, the input u is related to the second derivative of the output y (see Eq. (2.13)), hence the relative degree is 2.

2.3.2 Flat under-actuated systems

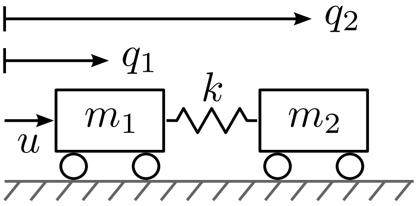


Figure 2.3: Flat under-actuated cart system with two dofs q_1 and q_2, connected using a linear spring with stiffness k. The system is actuated using one input u.

Let us now consider the flexible cart system shown in Fig. 2.3: two masses m_1 and m_2 connected using a localized linear spring element of stiffness k. The system is described by two dofs q_1 and q_2 and only the first dof q_1 is actuated using the input u, hence the system is under-actuated. The output is $y = q_2$ and has to follow the scalar prescribed trajectory $y_{\text{presc}}(t)$. The inverse dynamics problem is expressed as

$$m_1\ddot{q}_1 + k(q_1 - q_2) = u \tag{2.14}$$

$$m_2\ddot{q}_2 - k(q_1 - q_2) = 0 \tag{2.15}$$

$$q_2 = y_{\text{presc}}(t) \tag{2.16}$$

Substituting the servo-constraint given by Eq. (2.16) and its derivatives into the other two equations, the scalar input u_{FF} can be directly computed as

$$u_{\mathrm{FF}} = \frac{m_1 m_2}{k} y^{[4]}_{\mathrm{presc}} + (m_1 + m_2)\ddot{y}_{\mathrm{presc}} \tag{2.17}$$

Therefore, the output q_2 is a flat output of the system: the input effort and the dynamics of q_1 are determined algebraically by y_{presc} and its derivatives up to the fourth order. As a consequence, this system has relative degree 4.

Also, matrices $\mathbf{M}$, $\mathbf{D}$ and $\mathbf{A}$ are

$$\mathbf{M} = \begin{bmatrix} m_1 & 0 \\ 0 & m_2 \end{bmatrix} \quad ; \quad \mathbf{D} = \begin{bmatrix} 0 & 1 \end{bmatrix} \quad ; \quad \mathbf{A} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

and the matrix given by Eq. (2.7) is singular meaning that the index is not 3 (it can be shown that the index is actually 5 in this case).

2.3.3 Systems with minimum phase internal dynamics

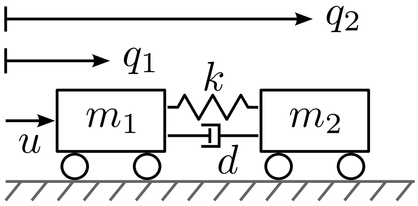


Figure 2.4: Minimum phase under-actuated cart system with two dofs q_1 and q_2, connected using a linear spring-damper element with stiffness k and damping d. The system is actuated using one input u.

Many systems are not differentially flat and, in such cases, the internal dynamics of the system needs to be solved. To illustrate this concept, consider the previous linear cart system where the two dofs q_1 and q_2 are additionally coupled with a damper element d. The equations governing the internal dynamics of the whole system now become

$$m_1\ddot{q}_1 + d(\dot{q}_1 - \dot{q}_2) + k(q_1 - q_2) = u \tag{2.18}$$

$$m_2\ddot{q}_2 - d(\dot{q}_1 - \dot{q}_2) - k(q_1 - q_2) = 0 \tag{2.19}$$

$$q_2 = y_{\mathrm{presc}}(t) \tag{2.20}$$

The feedforward input u_{FF} can be expressed as

$$u_{\mathrm{FF}} = m_1\ddot{q}_1 + m_2\ddot{y}_{\mathrm{presc}} \tag{2.21}$$

where the unknown dynamics of q_1 affects the input and needs to be solved. The equation governing this dynamics reads

$$d\dot{q}_1 + kq_1 = m_2\ddot{y}_{\mathrm{presc}} + d\dot{y}_{\mathrm{presc}} + ky_{\mathrm{presc}} \tag{2.22}$$

One can see that the first order dynamics $\dot{q}_1$ expressed by Eq. (2.22) involves the second derivative $\ddot{y}$ of the output. Since the input u_{FF} requires the second order dynamics $\ddot{q}_1$ (see Eq. (2.21)), one needs to derive the dynamics of Eq. (2.22) once more. As a result, the input is related to the third derivative $y^{[3]}$ of the output and the relative degree of this system is 3. Regarding the differential index of the resulting DAE, the matrix given by Eq. (2.7) is the same as for the previous flat system and is therefore singular. It can be shown that the index is 4 in this case.

Analyzing the generalized eigenvalues of Eq. (2.9) is equivalent to finding the root of the left hand side of Eq. (2.22) which is $-k/d$. This root has a negative real part for $d, k > 0$ and the zero dynamics of the system is therefore stable; one can say that the internal dynamics is minimum phase.

Consequently, a direct integration of the internal dynamics from an initial condition will result in a bounded solution for the input u_{FF}. For example, with the zero dynamics i.e. $y_{\mathrm{presc}} = \dot{y}_{\mathrm{presc}} = \ddot{y}_{\mathrm{presc}} = 0$, a solution of Eq. (2.22) for q_1 can be expressed in the form of

$$q_1(t) = Ae^{-at} \tag{2.23}$$

with $a = k/d$. The A constant can be determined by one initial condition, e.g. an initial velocity in the system expressed as $\dot{q}_1(0) = 1$, which leads to

$$A = -\frac{1}{a} \tag{2.24}$$

As a result, the solution for the feedforward input u_{FF} of Eq. (2.21) in the zero dynamics is

$$u_{\mathrm{FF}}(t) = -m_1 a e^{-at} \tag{2.25}$$

which fades out as time goes forward. Intuitively, if one wishes to compute the input u_{FF} that maintains the output y at zero, a bounded solution can be found by forward time integration although an initial perturbation e.g. a small initial velocity, is present in the model.

2.3.4 Systems with non-minimum phase internal dynamics

In contrast to minimum phase systems, a non-minimum phase system is a system with unstable internal dynamics. Let us now consider an under-actuated planar double pendulum depicted in Fig. 2.5. It is composed of two identical masses m linked through two identical mass-less rigid links of length l. The first link is actuated by the input u acting on the coordinate q_1 of the first hinge joint. The second link is connected to the first link through a passive hinge with stiffness k. The output $y = q_2$ is the absolute orientation of the second link.

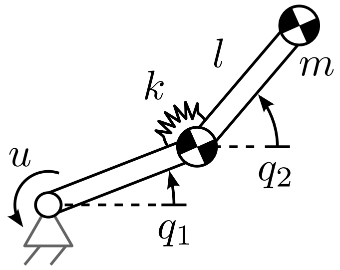


Figure 2.5: Non-minimum phase under-actuated pendulum with two dofs q_1 and q_2, connected using a linear spring with stiffness k. The system is actuated using one input u.

The linearized internal dynamics of the system for $(q_1 - q_2) \ll 1$ is governed by the following set of equations (see A.3)

$$2ml^2\ddot{q}_1 + ml^2\ddot{q}_2 - k(q_2 - q_1) = u \tag{2.26}$$

$$ml^2\ddot{q}_1 + ml^2\ddot{q}_2 + k(q_2 - q_1) = 0 \tag{2.27}$$

$$q_2 = y_{\text{presc}}(t) \tag{2.28}$$

Again, substituting the servo constraint (Eq. (2.28)) into the first equation leads to the expression of the forward input torque u_{FF} as

$$u_{\text{FF}} = 3ml^2\ddot{q}_1 + 2ml^2\ddot{y}_{\text{presc}} \tag{2.29}$$

where the dynamics of $\ddot{q}_1$ given by

$$ml^2\ddot{q}_1 - kq_1 = -ml^2\ddot{y}_{\text{presc}} - ky_{\text{presc}} \tag{2.30}$$

still needs to be solved. As one can see from Eq. (2.30), this second order dynamics $\ddot{q}_1$ involves the second derivative $\ddot{y}$ of the output. Since the expression of the input u (see Eq. (2.29)) does not involve higher order derivatives, the relative degree between the input and the output is 2 in this case. Also, matrices **D** and **A** are the same as the flat and the minimum phase systems but the mass matrix **M** is now

$$\mathbf{M} = ml^2 \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix}$$

and has off diagonal coupling terms. As a result, the matrix given by Eq. (2.7) is non singular and the index of the DAE is 3.

In this case, the generalized eigenvalues of Eq. (2.9) are the roots of the second order left hand side of Eq. (2.30). The two roots are $+\sqrt{k/ml^2}$ and $-\sqrt{k/ml^2}$. Since one has a positive real part, the zero dynamics is unstable and the internal dynamics is non-minimum phase.

As a consequence, if a solution for u_{FF} is computed by integrating Eq. (2.30) from some initial conditions, an unbounded solution is found. As it is done for the previous example, with the zero dynamics i.e. $y_{\mathrm{presc}} = \dot{y}_{\mathrm{presc}} = \ddot{y}_{\mathrm{presc}} = 0$, a solution of Eq. (2.30) for q_1 can be expressed in the form of

$$q_1(t) = Ae^{-at} + Be^{at} \tag{2.31}$$

with $a = \sqrt{k/ml^2}$ in this case. The constants A and B can be determined by two initial conditions, e.g. an initial angle $q_1(0) = 0$ and an initial angular velocity $\dot{q}_1(0) = 1$, which leads to

$$\begin{aligned} A &= -\frac{1}{2a} \\ B &= \frac{1}{2a} \end{aligned} \tag{2.32}$$

Therefore, a solution for the feedforward input u_{FF} of Eq. (2.29) in the zero dynamics is

$$u_{\mathrm{FF}}(t) = 3ml^2\frac{a}{2}\left(e^{at} - e^{-at}\right) \tag{2.33}$$

which means that as time goes forward, the first term in Eq. (2.33) grows exponentially (while the second fades out). In other words, if one wishes to compute the input u_{FF} that maintains the output y at zero, a small perturbation of the model e.g. a small initial velocity, leads to a solution that grows unbounded rather quickly.

In order to find a bounded solution u_{FF} of the inverse dynamics problem, appropriate methods, such as the ones described in the introduction, have to be used.

2.4 Summary

In this chapter, the inverse dynamics problem is illustrated with some linear systems. The concept of differential index of the equations of motion and relative degree of a system are presented. For these simple systems, it is shown how the internal dynamics can affect the solution of the inverse dynamic problem.

These results can be generalized to more complex systems. In fact, the roots of the internal dynamics equations, or generalized eigenvalues of Eq. (2.9), represent the poles of the inverse dynamics transfer function around given equilibrium points. Each eigenvalue is related to an eigenvector: a stable eigenvalue is related to a vector $\boldsymbol{\zeta}_{\mathrm{s}}$, spanning in a so-called stable manifold of the linearized system while the unstable eigenvalue is related to a vector $\boldsymbol{\zeta}_{\mathrm{us}}$, spanning in a so-called unstable manifold of the linearized system.

From the simple examples of this chapter, one can observe that the relative degree for each input/output pair is one less that the index. This is true for many mechanical system but is not a general rule [27]. Furthermore, for linear systems, the relative degree is related to the number of poles of the inverse system. In fact, in that case, the relative degree is the difference between the number of poles and the number of zeros of the inverse dynamic transfer function. However, this may not be true for general MIMO non-linear systems. Also, the validity of large eigenvalues is questionable for the more complex spatial systems considered in this work. Indeed, the kinematic constraints used in the FEM model leads to infinite eigenvalues that may not correspond to real characteristics of the system [51]. As a result, an arbitrary tolerance to select the eigenvalues is set numerically and the number of poles of the system may be altered. For these reasons, the relative degree of the spatial systems presented in this work will not be discussed.

Chapter 3

Geometric modeling of flexible manipulators

(a) Schematic view. (b) Frames attached to each link.

Figure 3.1: Model of a serial rigid manipulator with two links.

In this chapter, the description of dynamic models of MBS using the local frame FEM approach is given. This approach is a general formulation that allows one to treat rigid links, flexible links or flexible joints as well as serial or parallel manipulators in one common framework. The first step in defining a FEM model is to describe the motion of its components. For a 3D rigid body, this motion is described by 4×4 transformation matrices $\mathbf{H}$ defined as

$$\mathbf{H} = \begin{bmatrix} \mathbf{R} & \mathbf{p} \\ \mathbf{0} & 1 \end{bmatrix}$$

Each matrix represents a frame attached to a body, as depicted in Fig. 3.1(b) for each rigid link, and involves a position vector $\mathbf{p} \in \mathbb{R}^3$ and a 3×3 rotation matrix $\mathbf{R}$. More particularly, such transformation matrices are elements of the so-called *special Euclidean group* $SE(3)$ which is a non-linear *matrix Lie group*. The properties of Lie groups can be studied using the theory of differential geometry. Then, practical tools allow one to derive the equations of motion of spatial MBS efficiently. The mathematical concepts presented in the following are developed with deeper explanations in reference [101]. More details about the formulation of the FEM approach in this framework can be found in [21, 107]. This chapter focuses on the material useful for the description of the inverse dynamics solver.

After introducing some fundamental tools about Lie groups, the description of the kinematics of a rigid body using transformation matrices is shown. Then the description of kinematic joints and flexible beams is given. Finally, the resulting FEM formulation for flexible MBS and a general procedure to solve the dynamics of such system are shown.

3.1 Lie group fundamentals

This section introduces some general and abstract concepts about Lie groups. Their application to the particular cases of the group of rotations $SO(3)$ and the group of rigid body motions $SE(3)$ is described in Section 3.2.

3.1.1 Definitions

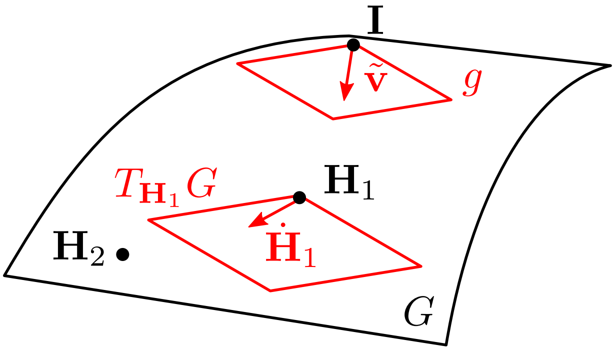


Figure 3.2: Representation of a Lie group as a non-linear space.

A *group* G is defined as a set of elements $\mathbf{H}$ with a composition rule, denoted as $\circ$, that respect the following axioms:

- closure: the composition of two elements $\mathbf{H}_1$ and $\mathbf{H}_2 \in G$ leads to an element of G i.e. $\mathbf{H}_1 \circ \mathbf{H}_2 = \mathbf{H}_3 \in G$.
- associativity: $(\mathbf{H}_1 \circ \mathbf{H}_2) \circ \mathbf{H}_3 = \mathbf{H}_1 \circ (\mathbf{H}_2 \circ \mathbf{H}_3)$.
- identity: there exist an element $\mathbf{I} \in G$ such that $\mathbf{H} \circ \mathbf{I} = \mathbf{I} \circ \mathbf{H} = \mathbf{H}$.

- invertibility: for each element $\mathbf{H} \in G$, there is an element $\mathbf{H}^{-1} \in G$ such that $\mathbf{H} \circ \mathbf{H}^{-1} = \mathbf{H}^{-1} \circ \mathbf{H} = \mathbf{I}$.

In addition to these axioms, a *matrix Lie group* has the structure of a differentiable manifold and each element of the group can be represented as a matrix. The composition and inversion of such group are smooth maps and are represented by the matrix product and the matrix inversion. One can represent such a group as a non-linear space as shown in Fig. 3.2.

3.1.2 Lie algebra

Let us consider a dynamic system whose configuration is represented by $\mathbf{H} \in G$. Its trajectory $\mathbf{H}(t)$, where t is the time, is thus a curve in G. The time derivative $\dot{\mathbf{H}}$ belongs to the tangent space at the element $\mathbf{H}$, i.e. $\dot{\mathbf{H}} \in T_{\mathbf{H}}G$. The Lie algebra g is the tangent space of the group G at the identity element $\mathbf{I}$, as shown in Fig. 3.2. This Lie algebra is a linear space and is therefore said to be isomorphic to $\mathbb{R}^k$, i.e. an element of this Lie algebra can be represented as a k component vector, with k being the dimension of the Lie algebra. Let us define a vector $\mathbf{v} \in \mathbb{R}^k$. This vector can be mapped to the Lie algebra g using the $\tilde{\bullet}$ operator. Given an element $\mathbf{H} \in G$ of the Lie group, the time derivative $\dot{\mathbf{H}}$ of this element is then conveniently defined using the Lie algebra element $\tilde{\mathbf{v}} \in g$ as

$$\dot{\mathbf{H}} = \mathbf{H}\tilde{\mathbf{v}} \tag{3.1}$$

Similarly, the variation $\delta\mathbf{H}$ of an element of G can be written as

$$\delta\mathbf{H} = \mathbf{H}\widetilde{\delta\mathbf{h}} \tag{3.2}$$

where $\delta\mathbf{h}$ represents an infinitesimal motion $\in \mathbb{R}^k$ that can be mapped to the Lie algebra as $\widetilde{\delta\mathbf{h}}$.

Furthermore, as shown in [107], when an element of a Lie group is represented by a matrix $\mathbf{H}$, its cross derivative is commutative which leads to

$$\delta(\dot{\mathbf{H}}) = (\delta\mathbf{H})^{\cdot} \tag{3.3}$$

Developing this property further (as detailed in A.1), one finds that

$$\delta(\tilde{\mathbf{v}}) - (\widetilde{\delta\mathbf{h}})^{\cdot} = \left[\tilde{\mathbf{v}}, \widetilde{\delta\mathbf{h}}\right] \tag{3.4}$$

in which the $[\bullet,\bullet]$ operator is called the *Lie bracket* and is defined as

$$[\bullet,\bullet] \;:\; g \times g \rightarrow g,\; \left[\tilde{\mathbf{v}}, \widetilde{\delta\mathbf{h}}\right] \mapsto \tilde{\mathbf{v}}\widetilde{\delta\mathbf{h}} - \widetilde{\delta\mathbf{h}}\tilde{\mathbf{v}} \tag{3.5}$$

Alternatively, Eq. (3.4) can be written in terms of vectors $\in \mathbb{R}^k$ as

$$\delta(\mathbf{v}) - (\delta\mathbf{h})^{\cdot} = \hat{\mathbf{v}}\delta\mathbf{h} \tag{3.6}$$

where the $\hat{\bullet}$ operator maps a vector of $\mathbb{R}^k$ into a $k \times k$ matrix.

Note It can be noted that another expression of the time derivative $\dot{\mathbf{H}}$ can also be found in mechanical text books. Indeed, instead of Eq. (3.1), one could also express the time derivative as $\dot{\mathbf{H}} = \tilde{\mathbf{v}}_R\mathbf{H}$ where the element of the Lie algebra $\tilde{\mathbf{v}}_R$ multiplies the element of the Lie group $\mathbf{H}$ from the right. However, it will be shown later that the left multiplication expressed in Eq. (3.1) leads to the so-called *local frame* representation. Therefore, in the remainder of this thesis, the latter expression is used.

3.1.3 Exponential map and tangent operator

The *exponential* mapping $\exp_G(\bullet)$ is an operator which maps an element $\widetilde{\mathbf{Q}}$ of the Lie algebra g to an element of the Lie group G,

$$\exp_G(\bullet) \;:\; g \rightarrow G,\; \widetilde{\mathbf{Q}} \mapsto \exp_G(\widetilde{\mathbf{Q}}) \tag{3.7}$$

The inverse of this mapping is called the *logarithmic* mapping $\log_G(\bullet)$,

$$\log_G(\bullet) \;:\; G \rightarrow g,\; \mathbf{H} \mapsto \log_G(\mathbf{H}) \tag{3.8}$$

These two mappings establish a relationship between the elements of non-linear matrix Lie group, such as $SE(3)$ or $SO(3)$, and their respective Lie algebra. They can be interpreted as a local parameterization of the elements of G: $\mathbf{H}$ can now be represented by an element of the Lie algebra or by a k-dimensional vector.

From the above definition, one can define a $\widetilde{\mathbf{Q}} \in g$ for any $\mathbf{H}_1, \mathbf{H}_2 \in G$ such that

$$\mathbf{H}_2 = \mathbf{H}_1 \exp_G(\widetilde{\mathbf{Q}}) \tag{3.9}$$

If $\mathbf{H}_1$ is constant in time, the derivative of $\mathbf{H}_2$ given by Eq. (3.1) can also be written

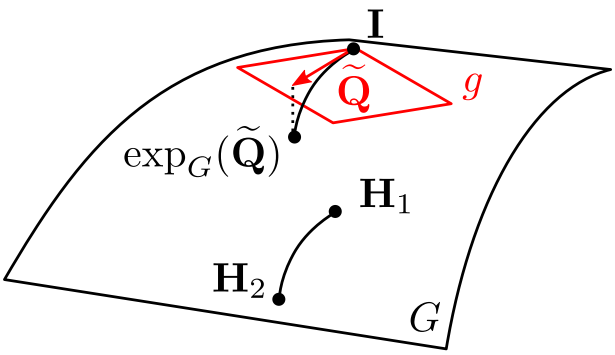


Figure 3.3: Mapping of an element of the Lie algebra g to an element of the Lie group G using the exponential map $\exp_G(\bullet)$.

as

$$\dot{\mathbf{H}}_2 = \mathbf{H}_2 \; \widetilde{\mathbf{T}(\mathbf{Q})\dot{\mathbf{Q}}} \tag{3.10}$$

with

$$\mathbf{v}_2 = \mathbf{T}(\mathbf{Q})\dot{\mathbf{Q}} \tag{3.11}$$

The so-called tangent operator $\mathbf{T}(\bullet)$ involved in the above expression represents the linearization of the exponential map and defines a mapping between a vector $\mathbf{v}$ and the derivative $\dot{\mathbf{Q}}$ of another vector. The detailed expressions of the exponential map, logarithmic map and tangent operator are given in A.2.

3.2 Kinematics of a rigid body

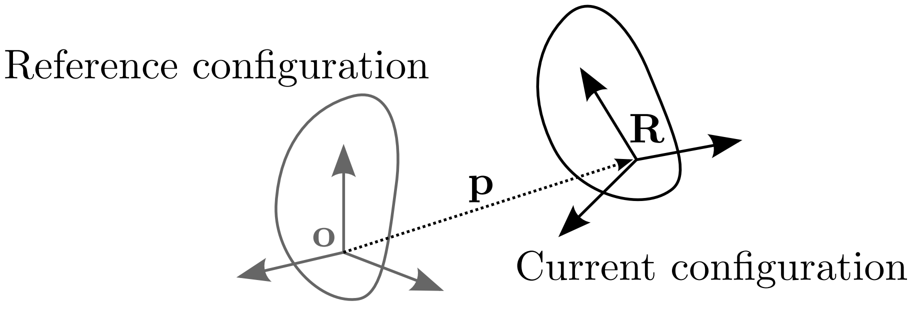


Figure 3.4: Kinematics of a rigid body.

Let us assume a general rigid body whose reference point is located at the origin $\mathbf{o}$ of a fixed inertial frame in its reference configuration. The configuration of this body can change and a current configuration is defined (see Fig 3.4). The orientation of this current configuration can be described by a rotation matrix $\mathbf{R} \in SO(3)$. The *special Orthogonal group* $SO(3)$ is the group of rotation matrices defined as

$$SO(3) = \{ \ \mathbf{R} \in \mathbb{R}^{3\times 3} \mid \mathbf{R}^T\mathbf{R} = \mathbf{I}, \ \det(\mathbf{R}) = +1 \ \} \tag{3.12}$$

with the matrix multiplication as composition rule. The Lie algebra associated to the $SO(3)$ group is denoted as $so(3)$ and has dimension $k = 3$. In addition, the position of the current configuration is described by a position vector $\mathbf{p} \in \mathbb{R}^3$. Eventually, the complete transformation from the reference configuration to the current configuration can be conveniently represented as a 4×4 transformation matrix $\mathbf{H}(\mathbf{p}, \mathbf{R})$ belonging to the *special Euclidean group* $SE(3)$ as follows

$$\mathbf{H}(\mathbf{p}, \mathbf{R}) \ = \begin{bmatrix} \mathbf{R} & \mathbf{p} \\ \mathbf{0} & 1 \end{bmatrix} \in SE(3) \tag{3.13}$$

The composition rule of the $SE(3)$ group is also the matrix multiplication. It can be noted that the Lie algebra of $SE(3)$, denoted as $se(3)$, has dimension $k = 6$. Therefore the representation using 4×4 transformation matrices is a 6-parameter transformation. The first three naturally result from the position vector $\mathbf{p}$. The last three, result from associating the 9 components of the rotation matrix $\mathbf{R}$ with the 6 orthonormality conditions $\mathbf{R}^T\mathbf{R} = \mathbf{I}$. This 4×4 matrix can also be seen as the description of a frame rigidly attached to the body. It can easily be shown that the inverse of such transformation matrix $\mathbf{H}^{-1}$ is

$$\mathbf{H}^{-1} = \begin{bmatrix} \mathbf{R}^T & -\mathbf{R}^T\mathbf{p} \\ \mathbf{0} & 1 \end{bmatrix} \tag{3.14}$$

From Eq. (3.1), the velocities of the body are expressed by the time derivative of a transformation matrix $\mathbf{H}$ as

$$\dot{\mathbf{H}} = \mathbf{H}\tilde{\mathbf{v}} = \begin{bmatrix} \mathbf{R} & \mathbf{p} \\ \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{v}}_R & \mathbf{v}_p \\ \mathbf{0} & 0 \end{bmatrix} \tag{3.15}$$

with $\tilde{\mathbf{v}}(\mathbf{v}_R, \mathbf{v}_p) \in se(3)$. This equation is also called the *compatibility equation.* Decomposing Eq. (3.15) into the translational velocity $\dot{\mathbf{p}}$ and the rotational velocity $\dot{\mathbf{R}}$ leads to

$$\begin{aligned} \dot{\mathbf{p}} &= \mathbf{R}\mathbf{v}_p \\ \dot{\mathbf{R}} &= \mathbf{R}\tilde{\mathbf{v}}_R \end{aligned} \tag{3.16}$$

From Eq. (3.16), one can see that $\mathbf{v}_p = \mathbf{R}^T\dot{\mathbf{p}} \in \mathbb{R}^3$ is the translational velocity vector of the current configuration with respect to the fixed inertial frame, expressed in the current frame $\mathbf{H}(\mathbf{p},\mathbf{R})$. Equivalently, the angular velocity vector $\mathbf{v}_R \in \mathbb{R}^3$ expresses the rotational velocity in the current frame. As a consequence, the vector $\mathbf{v}(\mathbf{v}_R,\mathbf{v}_p) \in \mathbb{R}^6$ is considered as the *local frame* representation of the velocities.

The 3 component vector $\mathbf{v}_R$ is represented in the Lie algebra $so(3)$ by a skew-symmetric matrix as

$$\tilde{\mathbf{v}}_R = \begin{bmatrix} 0 & -\mathbf{v}_{R3} & \mathbf{v}_{R2} \\ \mathbf{v}_{R3} & 0 & -\mathbf{v}_{R1} \\ -\mathbf{v}_{R2} & \mathbf{v}_{R1} & 0 \end{bmatrix} \in so(3) \tag{3.17}$$

In contrast, the 6 component vector $\mathbf{v}$ is represented in the Lie algebra $se(3)$ by a 4×4 matrix, as in Eq. (3.15). These notations are similar to references [51, 70, 84, 108].

3.3 Dynamics of a rigid body

The Hamilton's principle states that the trajectory of a conservative system between two time instants t_i and t_f is such that its action integral is stationary provided that its initial and final configurations are fixed, mathematically

$$\delta\left(\int_{t_i}^{t_f} (\mathcal{K} - \mathcal{V}_{\text{ext}})\ dt\right) = 0 \tag{3.18}$$

where $\mathcal{K}$ and $\mathcal{V}_{\text{ext}}$ are the kinetic and potential energy of the system respectively. For a rigid body, the kinetic energy is given as

$$\mathcal{K} = \frac{1}{2}\mathbf{v}^T\mathbf{M}\mathbf{v} \tag{3.19}$$

where $\mathbf{M}$ is a constant 6×6 matrix of the form

$$\mathbf{M} = \begin{bmatrix} m\mathbf{I}_{3\times 3} & \mathbf{0} \\ \mathbf{0} & \mathbf{J} \end{bmatrix} \tag{3.20}$$

with m the mass of the body and $\mathbf{J} \in \mathbb{R}^{3\times 3}$ its rotation inertia defined at the center of mass of the body. The fact that $\mathbf{M}$ is constant is a direct consequence of the local frame representation of the velocities shown previously. It is further assumed that the potential field is a function of $\mathbf{H}$.

The variation of the kinetic energy $\delta\mathcal{K}$ is expressed using Eq. (3.6) and the infinitesimal motion $\delta\mathbf{h}$ of Eq. (3.2), leading to

$$\delta\left(\int_{t_i}^{t_f} \mathcal{K}\, dt\right) = \int_{t_i}^{t_f} \left(\left(\delta\dot{\mathbf{h}}\right)^T + \delta\mathbf{h}^T\hat{\mathbf{v}}^T\right)\mathbf{M}\mathbf{v}\, dt \tag{3.21}$$

where the $\hat{\bullet}$ operator for the case of a vector $\mathbf{v}(\mathbf{v}_R, \mathbf{v}_p) \in \mathbb{R}^6$ is expressed as

$$\hat{\mathbf{v}} = \begin{bmatrix} \tilde{\mathbf{v}}_R & \tilde{\mathbf{v}}_p \\ \mathbf{0} & \tilde{\mathbf{v}}_R \end{bmatrix} \tag{3.22}$$

Integrating by parts the left hand side of Eq. (3.21) leads to

$$\delta\left(\int_{t_i}^{t_f} \mathcal{K}\, dt\right) = \left[\delta\mathbf{h}^T\mathbf{M}\mathbf{v}\right]_{t_i}^{t_f} - \int_{t_i}^{t_f} \delta\mathbf{h}^T\left(\mathbf{M}\dot{\mathbf{v}} - \hat{\mathbf{v}}^T\mathbf{M}\dot{\mathbf{v}}\right)\, dt \tag{3.23}$$

Since the configuration $\mathbf{H}$ at the initial and final time is fixed, the infinitesimal motion $\delta\mathbf{h}$ in the first term on the left hand side of Eq. (3.23) vanishes and

$$\delta\mathcal{K} = -\delta\mathbf{h}^T\mathbf{g}_{\text{ine}}(\mathbf{v}, \dot{\mathbf{v}}) = -\delta\mathbf{h}^T\left(\mathbf{M}\dot{\mathbf{v}} - \hat{\mathbf{v}}^T\mathbf{M}\mathbf{v}\right) \tag{3.24}$$

The variation of the potential energy is expressed based on the external forces $\mathbf{g}_{\text{ext}}(\mathbf{H})$ in the system as

$$\delta\mathcal{V}_{\text{ext}} = \delta\mathbf{h}^T\mathbf{g}_{\text{ext}}(\mathbf{H}) \tag{3.25}$$

Inserting Eqs. (3.24) and (3.25) in Hamilton's principle Eq. (3.18) and since the stationary property must be true for any arbitrary motion $\delta\mathbf{h}$, the equation of motion of the rigid body follows

$$\mathbf{g}_{\text{ine}}(\mathbf{v}, \dot{\mathbf{v}}) = \mathbf{g}_{\text{ext}}(\mathbf{H}) \tag{3.26}$$

i.e. the inertia forces and external forces are at equilibrium.

3.4 Kinematic constraints

Up to now, the spatial motion of rigid bodies was considered. It was shown that such 3D motion can be conveniently represented by 4×4 transformation matrices $\in SE(3)$. In general MBS, bodies are connected to one another using various types of joints. These joints constrain each bodies to certain relative motions. Depending on the type of joint that is considered, the number of dofs of the joint is $m \leq 6$. Mathematically, such relative motions can be described by a sub-space of the $SE(3)$

group. This leads to the concept of *subgroups* of $SE(3)$ [101, 107]. In the present formulation, the relative motion of a joint J is represented by an element $\mathbf{H}_J$ that belongs to $SE(3)$. However, the underlying Lie algebra associated to this element has dimension $k = m$ and is a sub-space of $se(3)$. Therefore, one can define some elements $\tilde{\mathbf{v}}_j$ and $\widetilde{\delta\mathbf{h}}_j$ of this Lie algebra such that $\mathbf{v}_j \in \mathbb{R}^m$ and $\delta\mathbf{h}_j \in \mathbb{R}^m$. One can further define a $6{\times}m$ matrix $\mathbf{A}_J$ describing a linear map between elements $\tilde{\mathbf{v}}_j$ of the Lie algebra of the subgroup and elements $\tilde{\mathbf{v}}_J$ of the Lie algebra $se(3)$ as $\mathbf{v}_J = \mathbf{A}_J\mathbf{v}_j$. We have:

$$\dot{\mathbf{H}}_J = \mathbf{H}_J\tilde{\mathbf{v}}_J = \mathbf{H}_J(\widetilde{\mathbf{A}_J\mathbf{v}_j}) \; ; \; \delta\mathbf{H}_J = \mathbf{H}_J\widetilde{\delta\mathbf{h}}_J = \mathbf{H}_J(\widetilde{\mathbf{A}_J\delta\mathbf{h}_j}) \tag{3.27}$$

In practice, the columns of the $\mathbf{A}_J$ matrix represent the axes of joint J in the local frame.

Considering a system with N bodies and n joints, the complete configurations $\mathbf{H}$ is

$$\mathbf{H} = \mathrm{diag}(\mathbf{H}_1, \cdots, \mathbf{H}_N, \mathbf{H}_{J,1}, \cdots, \mathbf{H}_{J,n}) \tag{3.28}$$

and the associated velocities $\mathbf{v}$ are

$$\mathbf{v} = \left[\mathbf{v}_1^T, \cdots, \mathbf{v}_N^T, \mathbf{v}_{j,1}^T, \cdots, \mathbf{v}_{j,n}^T\right]^T \tag{3.29}$$

It is important to note that the configurations $\mathbf{H}$ involve absolute variables of the nodes, such as $\mathbf{H}_N$, and relative variables of the kinematic joints, such as $\mathbf{H}_{J,n}$. As a result, the present approach is a so-called *mixed* formulation of the configurations.

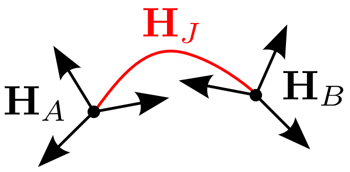


Figure 3.5: Relative configuration $\mathbf{H}_J$ between bodies A and B.

In the MBS model, such joints are described by a set of kinematic constraints $\mathbf{\Phi}$. Let us consider two bodies, A and B described respectively by their configuration $\mathbf{H}_A$ and $\mathbf{H}_B$. As shown in Fig. 3.5, the relative configuration between A and B is introduced by the element $\mathbf{H}_J$ associated to the joint J as

$$\mathbf{H}_B = \mathbf{H}_A\mathbf{H}_J \tag{3.30}$$

The kinematic constraint associated to joint J is therefore expressed using the matrix equation

$$\mathbf{H}_B^{-1}\mathbf{H}_A\mathbf{H}_J = \mathbf{I}_{4\times4} \tag{3.31}$$

Eq. (3.31) can be reformulated as a vectorial equation $\mathbf{\Phi}_J$ in the Lie algebra $se(3)$ using the logarithmic map $\log_{SE(3)}$ of the $SE(3)$ group (see Eq. (3.8)) as

$$\mathbf{\Phi}_J(\mathbf{H}_A, \mathbf{H}_B, \mathbf{H}_J) = \log_{SE(3)}(\mathbf{H}_B^{-1}\mathbf{H}_A\mathbf{H}_J) = \mathbf{0}_{6\times1} \tag{3.32}$$

In the present formulation, each joint therefore introduces $m \leq 6$ additional coordinates and 6 constraints to the system. It can also be useful to define the variation $\delta(\mathbf{\Phi}_J)$ of the constraint with respect to the configurations involved in joint J, hence

$$\delta(\mathbf{\Phi}_J) = \mathbf{B}(\mathbf{H}) \left[\delta\mathbf{h}_A^T \; \delta\mathbf{h}_B^T \; \delta\mathbf{h}_j^T\right]^T \tag{3.33}$$

where $\mathbf{B}$ is the $6\times(12+m)$ matrix of the constraint gradients of joint J.

This kinematic constraint equation can be included in the equation of motion of a rigid body Eq. (3.26) using the Lagrange multiplier method which leads to

$$\mathbf{g}_{\text{ine}}(\mathbf{v}, \dot{\mathbf{v}}) + \mathbf{B}^T(\mathbf{H})\boldsymbol{\lambda} = \mathbf{g}_{\text{ext}}(\mathbf{H}) \tag{3.34}$$

$$\mathbf{\Phi}(\mathbf{H}) = \mathbf{0} \tag{3.35}$$

in which $\boldsymbol{\lambda}$ are the Lagrange multipliers associated with the constraints $\mathbf{\Phi}$.

3.4.1 Example: the revolute joint

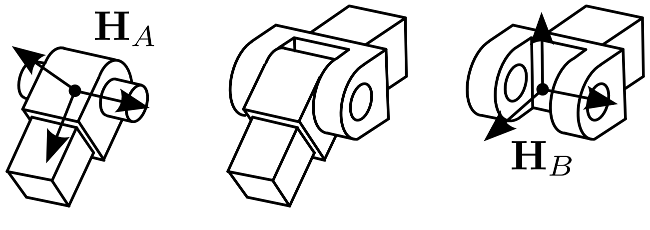


Figure 3.6: Revolute joint linking body A and body B.

To illustrate the above kinematic constraint formulation, let us consider the case of a revolute joint, also called hinge joint, as shown in Fig. 3.6. The revolute joint constrains two bodies A and B to have the same position i.e., $\mathbf{p}_A = \mathbf{p}_B$, but allows one rotation dof around its axis of rotation i.e., $\mathbf{R}_A \neq \mathbf{R}_B$. Therefore the dimension of the underlying Lie algebra is $m = 1$, i.e. $\mathbf{v}_j = \dot{\alpha}_J$ and $\delta\mathbf{h}_j = \alpha_J$ are scalars. Assuming the axis of the joint is the local y axis, α_J is the relative angle between

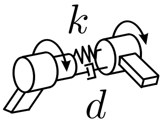


Figure 3.7: Revolute joint with internal stiffness k and viscous dissipation d.

body A and body B around the local y axis. The relative configuration $\mathbf{H}_J(\mathbf{p}_J, \mathbf{R}_J)$ corresponding to this revolute joint is defined by

$$\mathbf{R}_J = \begin{bmatrix} \cos\alpha_J & 0 & -\sin\alpha_J \\ 0 & 1 & 0 \\ \sin\alpha_J & 0 & \cos\alpha_J \end{bmatrix}; \quad \mathbf{p}_J = \mathbf{0} \tag{3.36}$$

and the $\mathbf{A}_J$ matrix in this case is

$$\mathbf{A}_J^T = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \tag{3.37}$$

It is interesting to note that the introduction of such additional coordinate leads to a straight forward expression of internal forces $\mathbf{g}_{\text{int,J}}$ inside such joint. For example, assuming the revolute joint has some internal stiffness k and viscous dissipation d (see Fig. 3.7), the internal forces $\mathbf{g}_{\text{int,J}}$ would be scalar and have the form

$$\mathbf{g}_{\text{int,J}} = (\alpha_J - \alpha_J^0)k + \dot{\alpha}_J d \tag{3.38}$$

Similarly, an external torque $\mathbf{g}_{\text{ext,J}} = \mathbf{u}$ produced by an actuator and applied on the α_J dof can be directly included in the model.

3.5 Flexible beam

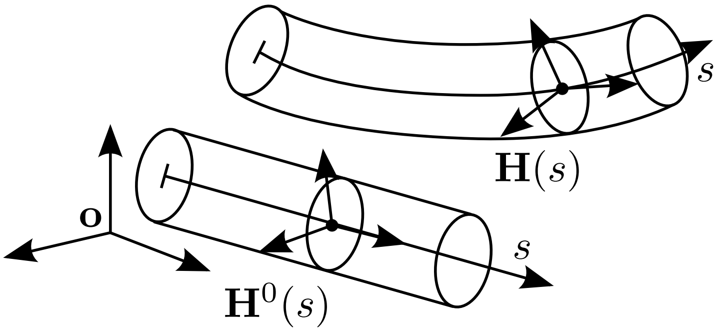


Figure 3.8: Initial and deformed configuration of a beam.

As this work focuses on flexibility effects in manipulators and more particularly in slender structural components, the flexible beam formulation presented in [109] is summarized in this section. Let us consider a beam of length L whose neutral axis is parameterized by a curvilinear coordinate $s \in [0; L]$. The cross section is assumed to remain plane and undeformed at any time. As a consequence, the cross section at point s can be considered as a rigid body described in its initial configuration by $\mathbf{H}^0(s)$ and in its current configuration by $\mathbf{H}(s)$ (as shown in Fig. 3.8).

3.5.1 Continuous formulation

By analogy with the rigid body description, the kinetic energy of the beam can be defined using the 6×1 velocity vector of the cross section $\mathbf{v}$ as

$$\mathcal{K} = \frac{1}{2} \int_L \mathbf{v}^T \mathbf{M}_b \mathbf{v} \; ds \tag{3.39}$$

where $\mathbf{M}_b$ is the cross section inertia matrix which is a constant matrix. The variation of the kinetic energy is expressed as

$$\delta\mathcal{K} = \int_L \delta\mathbf{h}^T \mathbf{g}_{\text{ine}}(\mathbf{v}, \dot{\mathbf{v}}) \; ds \tag{3.40}$$

where the beam inertia forces $\mathbf{g}_{\text{ine}}$ are defined in a similar way as in Eq. (3.24).

The potential energy of the beam $\mathcal{V} = \mathcal{V}_{\text{int}} + \mathcal{V}_{\text{ext}}$ has an internal contribution $\mathcal{V}_{\text{int}}$ from internal strains $\boldsymbol{\epsilon}$ and an external contribution $\mathcal{V}_{\text{ext}}$ from the external forces $\mathbf{g}_{\text{ext}}$.

The internal strains can be defined by first introducing the deformation gradient $\mathbf{f} \in \mathbb{R}^6$ related to the derivative $\mathbf{H}'(s)$ of the cross section configuration as

$$\mathbf{H}'(s) = \mathbf{H}(s)\tilde{\mathbf{f}}(s) \tag{3.41}$$

The strain $\boldsymbol{\epsilon} \in \mathbb{R}^6$ of the beam is then defined using the gradient $\mathbf{f}^0(s)$ at the initial configuration as

$$\boldsymbol{\epsilon}(s) = \mathbf{f}(s) - \mathbf{f}^0(s) \tag{3.42}$$

and is expressed in the local frame of the beam. The first component ϵ_1 of $\boldsymbol{\epsilon}$ represents the axial strain along the beam. The next two components ϵ_2 and ϵ_3 represent the shear strain in the cross section axes. The last three component represent the curvatures inside the beam: ϵ_4 is associated to the torsion of the beam while ϵ_5 and ϵ_6 represent the bending curvatures around the cross section axes.

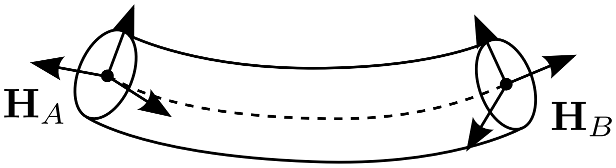


Figure 3.9: Flexible beam finite element between nodes A and B.

This strain measure leads to the expression of the internal energy as

$$\mathcal{V}_{\text{int}} = \frac{1}{2}\int_L \boldsymbol{\epsilon}^T \mathbf{K} \boldsymbol{\epsilon} \; ds \tag{3.43}$$

where $\mathbf{K}$ is the 6×6 stiffness of the beam defined by geometric and material characteristics. Assuming that the beam is initially straight with a Young modulus E, a cross section area A, a bulk modulus G and cross section moments of inertia I_1, I_2 and I_3, the stiffness matrix is $\mathbf{K} = \text{diag}(EA, GA, GA, GI_1, EI_2, EI_3)$. The variation of this internal energy leads, after some manipulations detailed in [109], to the expression of the internal resulting forces $\mathbf{g}_{\text{int}}$

$$\delta\mathcal{V}_{\text{int}} = \int_L \delta\boldsymbol{\epsilon}^T \mathbf{K} \boldsymbol{\epsilon} \; ds = \int_L \delta\mathbf{h}^T \mathbf{g}_{\text{int}} \; ds \tag{3.44}$$

The virtual work $\mathcal{V}_{\text{ext}}$ done by the external forces is defined as

$$\delta\mathcal{V}_{\text{ext}} = -\int_L \delta\mathbf{h}^T \mathbf{g}_{\text{ext}} \; ds \tag{3.45}$$

Using Hamilton's principle and combining Eqs. (3.40), (3.44) and (3.45), one obtains the dynamic equation of the beam

$$\mathbf{g}_{\text{ine}} + \mathbf{g}_{\text{int}} = \mathbf{g}_{\text{ext}} \tag{3.46}$$

3.5.2 Finite element discretization

In practice, approximation methods such as the FEM are used to find a numerical solution of Eq. (3.46). Such approximation method is based on a spatial discretization of the system into elements and nodes. The forces in the beam are then expressed in terms of forces acting on the nodes. Let us assume a beam finite element defined between two nodes A and B as shown in Fig. 3.9. The interpolated cross section configuration $\mathbf{H}(s)$ is defined using the exponential map (see Eq. (3.7)) as

$$\mathbf{H}(s) = \mathbf{H}_A \exp_{SE(3)}(\frac{s}{L}\tilde{\mathbf{d}}) \tag{3.47}$$

with $\mathbf{H}(s)|_{s=0} = \mathbf{H}_A$ and $\mathbf{H}(s)|_{s=L} = \mathbf{H}_B$. The *relative configuration vector* $\mathbf{d} \in \mathbb{R}^6$ is defined as

$$\mathbf{d} = \log_{SE(3)}(\mathbf{H}_A^{-1}\mathbf{H}_B) \tag{3.48}$$

The logarithmic mapping $\log_{SE(3)}(\bullet)$ is used to express the relative motion $\mathbf{H}_A^{-1}\mathbf{H}_B$, between nodes A and B, as a vector $\mathbf{d}$. In the discrete settings, the deformation gradient $\mathbf{f}$ of Eq. (3.41) is obtained as

$$\mathbf{f} = \frac{\mathbf{d}}{L} \tag{3.49}$$

and the strain $\boldsymbol{\epsilon}$ in the beam is then evaluated by comparing the relative configuration vector in the current configuration $\mathbf{d}$ and in the reference undeformed configuration $\mathbf{d}^0$

$$\boldsymbol{\epsilon} = \frac{\mathbf{d} - \mathbf{d}^0}{L} \tag{3.50}$$

In this formulation, one can observe that the strain $\boldsymbol{\epsilon}$ is constant over a beam element. In order to express the variation $\delta\boldsymbol{\epsilon}$ of the strain as a variation $\delta\mathbf{h}_{AB} = \left[\delta\mathbf{h}_A^T \ \delta\mathbf{h}_B^T\right]^T \in \mathbf{R}^{12}$ of the cross section configuration at nodes A and B, one can use the following relation

$$\delta\boldsymbol{\epsilon} = \frac{1}{L}\mathbf{P}(\mathbf{H})\delta\mathbf{h}_{AB} \tag{3.51}$$

where $\mathbf{P}(\mathbf{H})$ is a 6×12 matrix expressed as

$$\mathbf{P}(\mathbf{H}) = \left[-\mathbf{T}^{-1}(-\mathbf{d}) \quad \mathbf{T}^{-1}(\mathbf{d})\right] \tag{3.52}$$

From Eq. (3.44), the expression of the internal forces $\mathbf{g}_{\text{int}}$ acting on the nodes is therefore

$$\mathbf{g}_{\text{int}} = \mathbf{P}(\mathbf{H})^T\mathbf{K}\ \boldsymbol{\epsilon}(\mathbf{H}) \tag{3.53}$$

assuming the stiffness matrix $\mathbf{K}$ is constant over the beam element. An interesting property of the preceding expression can be pointed out: the internal forces $\mathbf{g}_{\text{int}}$ only depend on the relative configuration $\mathbf{H}_A^{-1}\mathbf{H}_B$ and not on the global motion of the beam. As a consequence, the internal forces benefits from reduced non-linearities.

For the kinetic energy of the beam, one can express the cross section velocity $\mathbf{v}(s)$ using the velocities $\mathbf{v}_{AB} = \left[\mathbf{v}_A^T \ \mathbf{v}_B^T\right]^T \in \mathbf{R}^{12}$ at nodes A and B using

$$\mathbf{v}(s) = \mathbf{Q}^{\star}(s, \mathbf{H})\mathbf{v}_{AB} \tag{3.54}$$

where $\mathbf{Q}^{\star}(s, \mathbf{H})$ is a 6×12 matrix expressed as

$$\mathbf{Q}^{\star}(\mathbf{H}) = \left[(\mathbf{I}_{6\times 6} - \tfrac{s}{L}\mathbf{T}(\tfrac{s}{L}\mathbf{d})\mathbf{T}^{-1}(\mathbf{d})) \quad \tfrac{s}{L}\mathbf{T}(\tfrac{s}{L}\mathbf{d})\mathbf{T}^{-1}(\mathbf{d}) \right] \tag{3.55}$$

and the inertia forces $\mathbf{g}_{\text{ine}}$ in the beam are then

$$\mathbf{g}_{\text{ine}} = \mathbf{M}(\mathbf{H})\dot{\mathbf{v}}_{AB} + \mathbf{C}(\mathbf{H}, \mathbf{v}_{AB})\mathbf{v}_{AB} \tag{3.56}$$

with $\mathbf{M}$ and $\mathbf{C}$ are 12×12 matrices defined as

$$\begin{aligned} \mathbf{M}(\mathbf{H}) &= \int_L \mathbf{Q}^{\star,T}\mathbf{M}_b\mathbf{Q}^{\star}\, ds && (3.57) \\ \mathbf{C}(\mathbf{H}, \mathbf{v}_{AB}) &= \int_L \mathbf{Q}^{\star,T}\left(\mathbf{M}_b\dot{\mathbf{Q}}^{\star} + \widehat{\mathbf{Q}^{\star}\mathbf{v}_{AB}}^{T}\mathbf{M}_b\mathbf{Q}^{\star}\right) ds && (3.58) \end{aligned}$$

in which the dependency on $\mathbf{H}$ and s of $\mathbf{Q}^{\star}(s, \mathbf{H})$ is not explicitly written for the sake of conciseness. As one can notice, $\mathbf{M}$ and $\mathbf{C}$ only depend on the relative configuration $\mathbf{H}_A^{-1}\mathbf{H}_B$ of the beam.

3.5.3 Internal dynamics and discretization artifacts

In this section, the stability of the internal dynamics described in Chapter 2 is briefly discussed for the present beam formulation. In [76], the continuous case of a Euler beam with transverse bending is considered. Its input and output are located at its opposite ends respectively and it is shown that the resulting continuous system is non-minimum phase. As it will be seen in Chapter 5, the present modeling approach of beams also leads to a non-minimum phase behavior of systems with mainly bending strains. The FEM discretization does not affect the stability characteristic of the system in this case.

However, it is interesting to note one particular case: a beam actuated axially at one end and with its output at the other end i.e. a bar system with axial deformations. In fact, it is shown in [76] that the continuous case of such bar system is minimum-phase. It is shown below that, for the discrete FEM case, this bar results in a non-minimum phase system.

Consider the simple planar two nodes beam system depicted in Fig. 3.10. The output $y = q_2$ is the axial position of the second node and the input u acts axially on the first node. As a result, the system is under-actuated and actually behaves like a bar with only axial strains. From the discrete model presented previously, the

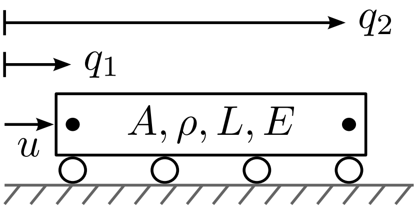


Figure 3.10: Non-minimum phase under-actuated system with two discrete dofs q_1 and q_2, connected using a flexible bar element. The system is actuated using one input u.

two nodes $\mathbf{H}_1$ and $\mathbf{H}_2$ are described as

$$\mathbf{H}_1 = \begin{bmatrix} 1 & 0 & 0 & q_1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} ; \quad \mathbf{H}_2 = \begin{bmatrix} 1 & 0 & 0 & q_2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.59}$$

and the relative configuration between the nodes is

$$\mathbf{H}_1^{-1}\mathbf{H}_2 = \begin{bmatrix} 1 & 0 & 0 & q_2 - q_1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.60}$$

Considering that the system has cross section A, density ρ, length L and Young modulus E, the inverse dynamics problem be expressed mathematically using only q_1 and q_2 as

$$\frac{m}{3}\ddot{q}_1 + \frac{m}{6}\ddot{q}_2 + \frac{EA}{L}(q_1 - q_2) = u \tag{3.61}$$

$$\frac{m}{6}\ddot{q}_1 + \frac{m}{3}\ddot{q}_2 - \frac{EA}{L}(q_1 - q_2) = 0 \tag{3.62}$$

$$q_2 = y_{\text{presc}}(t) \tag{3.63}$$

with $m = \rho AL$ is the mass of the bar. As described in Chapter 2, these equations govern the internal dynamics of the system and one can therefore be interested in its stability. Similarly to the developments done in Chapter 2, it can be shown that the system has two eigenvalues $+\sqrt{6EA/mL}$ and $-\sqrt{6EA/mL}$. As a result of the positive real eigenvalue, this discretized bar system is non-minimum phase. Unlike the continuous bar case [76], the discretization of the bar introduces an artifact that alters the stability characteristic of the system.

This artifact appears in the particular case of a bar with axial actuation. The simulation and experimental cases that are considered in the following chapters mainly deal with flexible systems in bending. In this work, it is therefore assumed that the artifact described in this section does not alter the stability behavior of the systems considered here. The validity of using such discrete FEM approach to model flexible manipulators is not discussed in this work.

3.6 Dynamics of a MBS

Eventually, one arrives at the description of a MBS which usually involves several rigid bodies, flexible bodies and kinematic constraints. The dynamics of such system can be expressed by combining Eq. (3.15), Eq. (3.34-3.35) and Eq. (3.46) which leads to

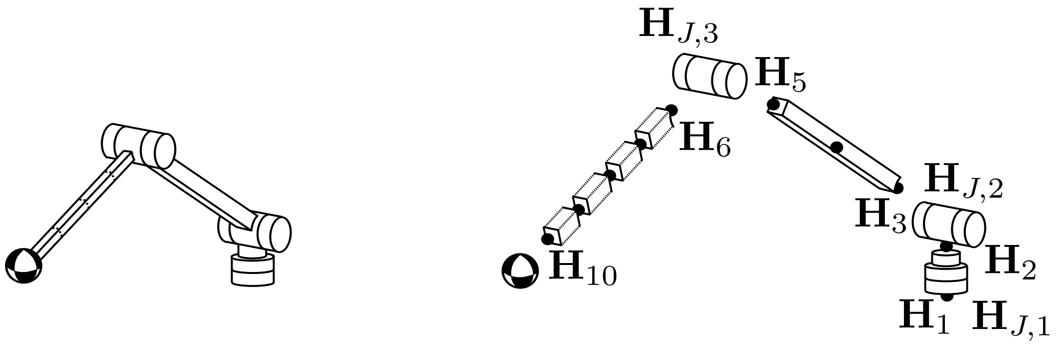
$$\begin{aligned}\dot{\mathbf{H}}_I &= \mathbf{H}_I\tilde{\mathbf{v}}_I &(3.64)\\ \mathbf{M}(\mathbf{H})\dot{\mathbf{v}}+\mathbf{g}(\mathbf{H},\mathbf{v})+\mathbf{B}(\mathbf{H})^T\boldsymbol{\lambda} &= \mathbf{A}\mathbf{u}(t) &(3.65)\\ \boldsymbol{\Phi}(\mathbf{H}) &= \mathbf{0} &(3.66)\end{aligned}$$

where the dependency on the acceleration $\dot{\mathbf{v}}$ is shown explicitly. In this expression, $\mathbf{g}$ includes the complementary inertia forces, external forces and internal forces arising from flexible components. As control problem are considered in this work, the term $\mathbf{A}\mathbf{u}(t)$ represent the actuation efforts in the system as described in Chapter 2. As shown in the preceding sections, the velocity vector and the deformations are expressed in the local frame of the body and as a result the internal forces $\mathbf{g}_{\text{int}}$ exhibit reduced non-linearities. The use of relative motions to describe the kinematic constraints $\boldsymbol{\Phi}$ also reduces the non-linearities in the latter.

One can note that the above equations are a set of differential algebraic equations (DAE) on a Lie group. The index of such DAE is 3 [27], i.e. in order to have ordinary differential equations (ODE), one needs to differentiate the algebraic constraints Eq. (3.66) three times. In the present formulation, such DAE on a Lie group can be solved without reducing its index using a generalized-α scheme [23] summarized in Section 3.7.

3.6.1 Example

To illustrate the construction of the model, an example of a 3 dof spatial serial manipulator is considered and shown in Fig. 3.11(a). This example is studied later



(a) Schematic view.

(b) Exploded view with nodes.

Figure 3.11: Example of a flexible serial arm with a point mass end-effector.

on in Chapter 5. It is a flexible manipulator composed of two links: the first one, connected to the base, is rigid while the second, connected to the end-effector, is flexible. Three actuated revolute joints allow the structure to move in 3D space. To construct the FEM model, the manipulator is discretized in space using nodes I, which are shown in the exploded view in Fig. 3.11(b). The position and orientation of each node I of the finite element mesh is represented as a transformation matrix $\mathbf{H}_I$ with a rotation $\mathbf{R}_I \in SO(3)$ and a position $\mathbf{p}_I \in \mathbb{R}^3$ (see Eq. (3.13)) with $I = 1, \cdots, N$ and $N = 10$ is the number of nodes of the mesh in this case. Additionally, the present system has $n = 3$ revolute joints with transformations $\mathbf{H}_{J,n}$. Similarly to Eq. (3.28) for multiple rigid bodies, the nodal configuration $\mathbf{H}$ of the complete system can then be represented as a block diagonal matrix that gathers each nodal variable and joint transformation

$$\mathbf{H} \; = \; \operatorname{diag}(\mathbf{H}_1, \cdots, \mathbf{H}_{10}, \mathbf{H}_{J,1}, \cdots, \mathbf{H}_{J,3}) \tag{3.67}$$

and the velocities $\mathbf{v}$ from Eq. (3.29) would be

$$\mathbf{v} = \left[\mathbf{v}_1^T, \cdots, \mathbf{v}_{10}^T, \dot{\alpha}_{J,1}, \dot{\alpha}_{J,2}, \dot{\alpha}_{J,3}\right]^T \tag{3.68}$$

3.7 Dynamic simulations and time discretization

Once the MBS is discretized in space, its dynamic behavior can be analyzed during a given time frame. Eqs. (3.64-3.66) can be integrated in time from given initial conditions to find a solution for $\mathbf{H}$, $\mathbf{v}$, $\dot{\mathbf{v}}$, $\boldsymbol{\lambda}$. This integration process can be done numerically by a time discretization of the dynamic equations over a finite number s of time steps. The extension of the classical generalized-α scheme to solve DAE on a Lie group [23] with a second-order convergence is summarized here.

The equations of motion in DAE form are written at time step $k+1$ as

$$\mathbf{H}_I^{k+1} = \mathbf{H}_I^k \exp(\widetilde{\mathbf{Q}_I^k}) \tag{3.69}$$

$$\mathbf{M}^{k+1}\dot{\mathbf{v}}^{k+1} + \mathbf{g}(\mathbf{H}^{k+1}, \mathbf{v}^{k+1}) + \mathbf{B}^{k+1,T}\boldsymbol{\lambda}^{k+1} = \mathbf{A}\mathbf{u}^{k+1} \tag{3.70}$$

$$\mathbf{\Phi}(\mathbf{H}^{k+1}) = \mathbf{0} \tag{3.71}$$

where $\widetilde{\mathbf{Q}}$ is an incremental configuration that belongs to the Lie algebra of $SE(3)$. The time integration formulas are used to compute the evolution of the variables for each time step k,

$$\mathbf{Q}^k = h\mathbf{v}^k + (\frac{1}{2} - \beta)h^2\mathbf{a}^k + \beta h^2 \mathbf{a}^{k+1} \tag{3.72}$$

$$\mathbf{v}^{k+1} = \mathbf{v}^k + (1-\gamma)h\mathbf{a}^k + \gamma h \mathbf{a}^{k+1} \tag{3.73}$$

$$(1-\alpha_m)\mathbf{a}^{k+1} = (1-\alpha_f)\dot{\mathbf{v}}^{k+1} + \alpha_f \dot{\mathbf{v}}^k - \alpha_m \mathbf{a}^k \tag{3.74}$$

where h is the time step size, $\mathbf{a}$ is considered as an acceleration-like variable of the algorithm, α_m, α_f, β and γ are integration parameters. The latter parameters can be defined to reach a chosen value of the spectral radius ρ,

$$\alpha_m = \frac{2\rho - 1}{\rho + 1} \quad ; \quad \alpha_f = \frac{\rho}{\rho+1} \quad ; \quad \beta = \frac{1}{(\rho+1)^2} \quad ; \quad \gamma = \frac{3-\rho}{2(\rho+1)}$$

The spectral radius ρ defines whether no numerical damping of the high frequency content ($\rho = 1$), maximum numerical damping ($\rho = 0$) or some numerical damping is considered $\rho \in \,]0;1[$.

Since the equations of motion are non-linear, an iterative Newton procedure based on a linearization of the equations is used. For this purpose, one defines the residual $\mathbf{r}^k$ of the system as the left hand side of Eq. (3.70). Also, a correction $\Delta\mathbf{H}$ of the configurations is defined as

$$\Delta\mathbf{H}^{k+1} = \mathbf{H}^{k+1}\widetilde{\Delta\mathbf{h}^k} \tag{3.75}$$

and similarly to Eq. (3.11), the tangent operator $\mathbf{T}$ is used to write the correction $\Delta\mathbf{h}^k$ as a correction $\Delta\mathbf{Q}^k$ of the incremental configurations, leading to

$$\Delta\mathbf{h}^k = \mathbf{T}(\mathbf{Q}^k)\Delta\mathbf{Q}^k \tag{3.76}$$

The linearization of Eqs. (3.70-3.71) leads to

$$
\begin{aligned}
d\mathbf{r}^k \cdot \Delta\mathbf{h}^k &= \mathbf{K}^k\mathbf{T}\Delta\mathbf{Q}^k && (3.77)\\
d\mathbf{r}^k \cdot \Delta\mathbf{v}^k &= \mathbf{C}^k \mathrm{d}\mathbf{v}^k && (3.78)\\
d\mathbf{r}^k \cdot \Delta\dot{\mathbf{v}}^k &= \mathbf{M}^k \mathrm{d}\dot{\mathbf{v}}^k && (3.79)\\
d\mathbf{r}^k \cdot \Delta\boldsymbol{\lambda}^k &= \mathbf{B}^{k,T} \mathrm{d}\boldsymbol{\lambda}^k && (3.80)\\
d\boldsymbol{\Phi}^k \cdot \Delta\mathbf{h}^k &= \mathbf{B}^k\mathbf{T}\Delta\mathbf{Q}^k && (3.81)
\end{aligned}
$$

Matrices $\mathbf{K}^k$, $\mathbf{C}^k$ and $\mathbf{M}^k$ are the tangent stiffness, tangent damping and tangent mass matrices. From Eqs. (3.72-3.74), the corrections $\Delta\mathbf{v}^k$ and $\Delta\dot{\mathbf{v}}^k$ of the velocities and acceleration can be written as

$$
\begin{aligned}
\Delta\mathbf{v}^k &= \gamma'\Delta\mathbf{Q}^k && (3.82)\\
\Delta\dot{\mathbf{v}}^k &= \beta'\Delta\mathbf{Q}^k && (3.83)
\end{aligned}
$$

with $\beta' = (1-\alpha_m)/(\beta h^2(1-\alpha_f))$ and $\gamma' = \gamma/\beta h$.

At each Newton iteration, the corrections $\Delta\mathbf{Q}^k$ and $\Delta\boldsymbol{\lambda}^k$ of the incremental configuration and the Lagrange multipliers are computed by solving

$$
\mathbf{S}_T^k \begin{bmatrix} \Delta\mathbf{Q}^k \\ \Delta\boldsymbol{\lambda}^k \end{bmatrix} = - \begin{bmatrix} \mathbf{r}^{k,\star} \\ \boldsymbol{\Phi}^{k,\star} \end{bmatrix} \tag{3.84}
$$

with $\mathbf{r}^{k,\star}$ and $\boldsymbol{\Phi}^{k,\star}$ being the residue and constraint evaluated at the current iteration and the iteration matrix $\mathbf{S}_T^k$ being defined as

$$
\mathbf{S}_T^k = \begin{bmatrix} \beta'\mathbf{M}^k + \gamma'\mathbf{C}^k + \mathbf{K}^k\mathbf{T}(\mathbf{Q}^k) & \mathbf{B}^{k,T} \\ \mathbf{B}^k\mathbf{T}(\mathbf{Q}^k) & \mathbf{0} \end{bmatrix} \tag{3.85}
$$

Hence, at each iteration, some corrections on the configuration $\Delta\mathbf{Q}^k$, velocities $\Delta\mathbf{v}^k$, accelerations $\Delta\dot{\mathbf{v}}^k$ and Lagrange multipliers $\Delta\boldsymbol{\lambda}^k$ are computed from Eq. (3.84) and Eqs. (3.82-3.83) which eventually leads to the convergence of the solution for time step k. This process is summarized in Fig. 3.12. From reference [108], it can be noted that thanks to the local representation of the velocities and deformations in the present formulation, some parts of the iteration matrix $\mathbf{S}_T$ can be assumed constant during the iteration process. This can strongly reduce the computational costs of the direct dynamic simulation.

The complete development presented in this chapter is implemented in the

GECOS (GEometric toolbox for COnstrained mechanical Systems) development code. Based on the Matlab® environment, this open source code is licensed under the Apache License, Version 2.0.

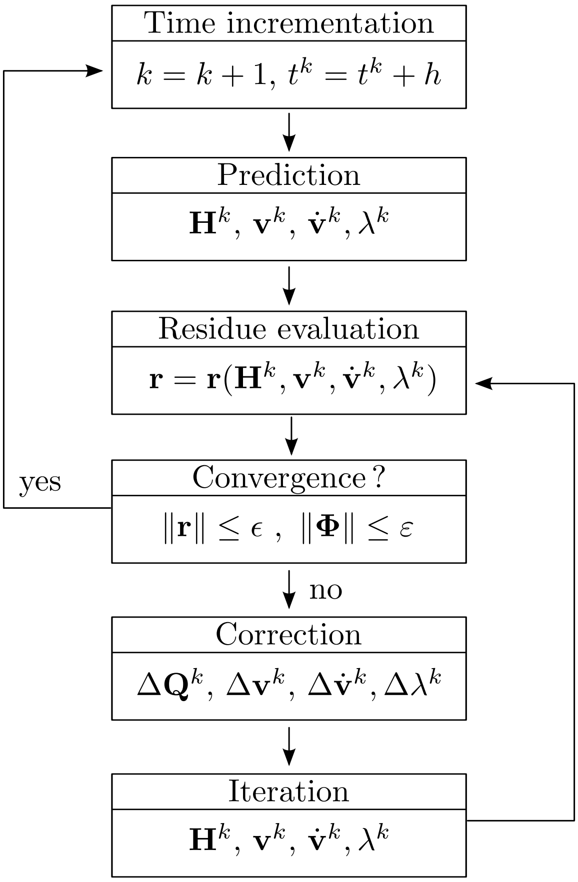


Figure 3.12: Iterative algorithm to solve the dynamic problem.

Chapter 4

Inverse dynamics of flexible manipulators

As stated in Section 1.1.2, given a prescribed trajectory $\mathbf{y}_{\text{presc}}$, the feedforward contribution of a robot controller generates reference actuator positions $\mathbf{q}_{\text{M,FF}}$ and actuator efforts $\mathbf{u}_{\text{FF}}$ that ideally lead to a perfect trajectory tracking of the output $\mathbf{y}$. Such reference inputs can be obtained by solving the inverse dynamics of the system. In this chapter, the inverse dynamics problem is stated for general flexible MBS modeled using a geometric FEM approach, as described in the previous chapter. The stable inversion method based on an optimization problem [4, 5, 6] and adapted to the $SE(3)$ formulation is then detailed. The present chapter is a central contribution of this work. To illustrate the methodology, the simple planar under-actuated cart example from [100] (Fig. 4.1) is considered.

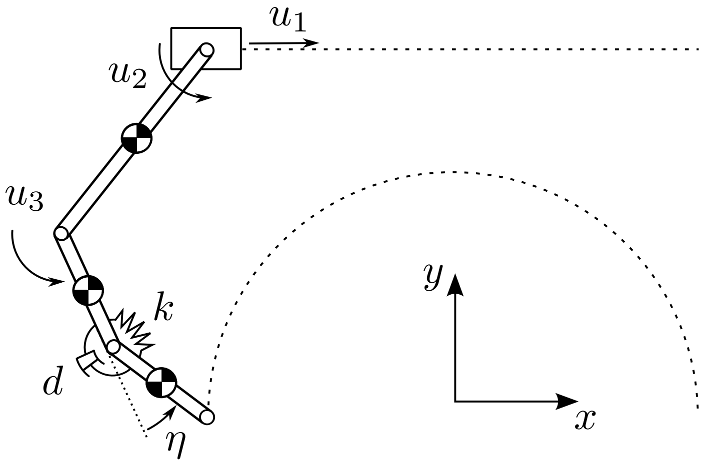


Figure 4.1: Planar under-actuated cart system with 4 rigid bodies and a passive joint.

This system is constructed with a rigid cart that can move along a linear axis in the x direction and a serial under-actuated manipulator mounted on it. The system is actuated with some input effort $\mathbf{u} = [u_1, u_2, u_3]$ acting on the cart and the two first joints of the manipulator respectively. The passive (non actuated) hinge is a flexible one with some stiffness k and damping d. The η coordinate describes the angle of this passive joint: the relative angle between the last two bodies of the arm. The cart is modeled as a point mass and the under-actuated manipulator is constructed with three rigid bodies.

4.1 Servo constraint in SE(3)

The *servo constraints* [14, 15, 99] of the inverse dynamics problem describe how the prescribed trajectory restricts a part of the system, i.e. the output. In this section, it is shown how such servo constraint can be expressed using the $SE(3)$ formulation. The discussion is first limited to the case where the output is part of a single node A of the system i.e. the overall number of servo constraint is $r \leq 6$.

Let us assume that there are r_A servo constraints on node A. The output $\mathbf{y}_A$ and the prescribed trajectory $\mathbf{y}_{\text{presc},A}$ at node A are therefore vectors $\in \mathbb{R}^{r_A}$: they belong to a sub-space of the configurations of the system, i.e. a subgroup of $SE(3)$. As it is done in Section 3.4 for the kinematic constraints, one can define a $6 \times r_A$ matrix $\mathbf{A}_A$ such that $\widetilde{\mathbf{A}_A \mathbf{y}_A} \in se(3)$ and $\widetilde{\mathbf{A}_A \mathbf{y}}_{\text{presc},A} \in se(3)$. Hence, the former is actually related to the configuration $\mathbf{H}_A(\mathbf{p}_A, \mathbf{R}_A)$ of node A through the exponential map $\exp_{SE(3)}(\bullet)$ as

$$\mathbf{H}_A = \exp_{SE(3)}(\widetilde{\mathbf{A}_A \mathbf{y}_A}) \in SE(3) \tag{4.1}$$

Similarly, the prescribed trajectory is mapped to an element $\mathbf{H}_{\text{presc}} \in SE(3)$ as

$$\mathbf{H}_{\text{presc}}(t) = \exp_{SE(3)}(\widetilde{\mathbf{A}_A \mathbf{y}}_{\text{presc},A}(t)) \tag{4.2}$$

and the servo constraint can then be expressed in matrix form as

$$\mathbf{H}_{\text{presc}}^{-1} \mathbf{H}_A = \begin{bmatrix} \mathbf{R}_{\text{presc}}^T \mathbf{R}_A & \mathbf{R}_{\text{presc}}^T (\mathbf{p}_A - \mathbf{p}_{\text{presc}}) \\ \mathbf{0} & 1 \end{bmatrix} = \mathbf{I} \tag{4.3}$$

where $\mathbf{p}_{\text{presc}}(t) \in \mathbb{R}^3$ and $\mathbf{R}_{\text{presc}}(t) \in SO(3)$ are interpreted as the position and orientation component of the prescribed trajectory.

Similarly to Eq. (3.16), the variation of the orientation part of Eq. (4.3) with respect to a variation $\delta\mathbf{h}_A$ of the configuration of node A is given by

$$\delta(\mathbf{R}_{\text{presc}}^T\mathbf{R}_A) = \mathbf{R}_{\text{presc}}^T\delta(\mathbf{R}_A) = \mathbf{R}_{\text{presc}}^T\mathbf{R}_A\ \widetilde{\delta\mathbf{h}}_{A,R} \tag{4.4}$$

where $\delta\mathbf{h}_{A,R} \in \mathbb{R}^3$ is the rotation component of $\delta\mathbf{h}_A$. The variation of the translation part of Eq. (4.3) leads to

$$\delta(\mathbf{R}_{\text{presc}}^T(\mathbf{p}_A - \mathbf{p}_{\text{presc}})) = \mathbf{R}_{\text{presc}}^T\delta(\mathbf{p}_A) = \mathbf{R}_{\text{presc}}^T\mathbf{R}_A\ \delta\mathbf{h}_{A,p} \tag{4.5}$$

where $\delta\mathbf{h}_{A,p} \in \mathbb{R}^3$ is the translation component of $\delta\mathbf{h}_A$.

Similarly to Eq. (3.32), the servo constraint on node A in matrix form given by Eq. (4.3) can be written as m vectorial equations $\mathbf{\Psi}_A(\mathbf{H}_A, t)$ using the logarithmic map $\log_{SE(3)}$ leading to

$$\mathbf{\Psi}_A(\mathbf{H}_A, t) = \mathbf{A}_A^T\log_{SE(3)}(\mathbf{H}_{\text{presc}}^{-1}\mathbf{H}_A) = \mathbf{0}_{r_A\times 1} \tag{4.6}$$

From the variation results above, one can express the variation of the servo constraints given by Eq. (4.6) as

$$\delta(\mathbf{\Psi}_A) = \mathbf{A}_A^T\mathbf{D}_A^\star\ \mathbf{T}^{-1}(\delta\mathbf{h}_A)\ \delta\mathbf{h}_A \tag{4.7}$$

where the gradient $\mathbf{D}_A^\star$ is a 6×6 matrix expressed as

$$\mathbf{D}_A^\star = \begin{bmatrix} \mathbf{R}_{\text{presc}}^T\mathbf{R}_A & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_{\text{presc}}^T\mathbf{R}_A \end{bmatrix} \tag{4.8}$$

The discussion is now generalized to the case where the output is defined as part of several nodes of the system. Physically, the number of servo constraint defined on a single node A is still limited to $r_A \leq 6$. However, assuming that the output is described over a number i of nodes of the system, the overall number of servo constraints $r = (r_A + \cdots + r_i)$ can now be greater than 6. The output $\mathbf{y} \in \mathbb{R}^r$ and the prescribed trajectory $\mathbf{y}_{\text{presc}} \in \mathbb{R}^r$ are then defined as

$$\mathbf{y} = \left[\mathbf{y}_A^T, \cdots, \mathbf{y}_i^T\right]^T \quad ; \quad \mathbf{y}_{\text{presc}} = \left[\mathbf{y}_{\text{presc},A}, \cdots, \mathbf{y}_{\text{presc},i}^T\right]^T \tag{4.9}$$

As a result, the servo constraints are described as a set of r vectorial equations

$$\mathbf{\Psi}(\mathbf{H}, t) = \begin{bmatrix} \mathbf{\Psi}_A(\mathbf{H}_A, t) \\ \vdots \\ \mathbf{\Psi}_i(\mathbf{H}_i, t) \end{bmatrix} = \mathbf{0}_{r\times 1} \tag{4.10}$$

and the complete gradient $\mathbf{D}$ of the servo constraints is a $r \times 6i$ bloc diagonal matrix and is expressed as

$$\mathbf{D} = \begin{bmatrix} \mathbf{A}_A^T \mathbf{D}_A^\star \mathbf{T}^{-1}(\delta \mathbf{h}_A) & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & \mathbf{A}_i^T \mathbf{D}_i^\star \mathbf{T}^{-1}(\delta \mathbf{h}_i) \end{bmatrix} \tag{4.11}$$

It can be noted that the above development is always valid when considering prescribed trajectory on the x, y and z position of a node. When considering trajectories on spatial orientations, the above development is valid only if all three components of the orientation are prescribed.

In the illustrative example of Fig. 4.1, the outputs of the system are $\mathbf{y} = [x_{\text{cart}}, x_{\text{eff}}, y_{\text{eff}}]$, the x position of the cart and the (x, y) coordinates of the manipulator's end-effector. From the discussion above, the output is therefore defined on two nodes of the system. One can then define two mappings $\mathbf{A}_{\text{cart}}$ and $\mathbf{A}_{\text{eff}}$ as

$$\mathbf{A}_{\text{cart}}^T = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad ; \quad \mathbf{A}_{\text{eff}}^T = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{4.12}$$

and $r = r_{\text{cart}} + r_{\text{eff}} = 3$. These outputs have to track the prescribed trajectory $\mathbf{y}_{\text{presc}}(t)$ represented in Fig. 4.1 in dotted lines.

4.2 Inverse dynamics problem

Generally speaking, the output $\mathbf{y} \in \mathbb{R}^r$ of a manipulator should follow a prescribed trajectory $\mathbf{y}_{\text{presc}}(t)$ thanks to r control inputs $\mathbf{u} \in \mathbb{R}^r$ acting on its joints. In inverse dynamics problems, a solution for the unknown inputs $\mathbf{u}$ must be found given the prescribed trajectory $\mathbf{y}_{\text{presc}}(t)$. With the present formulation, the inverse dynamics problem can be formulated mathematically based on the equations of the internal dynamics represented by Eqs. (3.64-3.66), recalled hereafter for clarity,

$$\begin{aligned} \dot{\mathbf{H}}_I &= \mathbf{H}_I \tilde{\mathbf{v}}_I && (4.13) \\ \mathbf{M}(\mathbf{H})\dot{\mathbf{v}} + \mathbf{g}(\mathbf{H}, \mathbf{v}) + \mathbf{B}(\mathbf{H})^T \boldsymbol{\lambda} &= \mathbf{A}\mathbf{u} && (4.14) \\ \boldsymbol{\Phi}(\mathbf{H}) &= \mathbf{0} && (4.15) \end{aligned}$$

supplemented by the additional servo constraints given by Eq. (4.10)

$$\boldsymbol{\Psi}(\mathbf{H}, t) = \mathbf{0}_{r \times 1} \qquad (4.16)$$

As stated in the previous chapter, Eqs. (4.13-4.15) are a set of index 3 DAE on a Lie group. However, with the additional servo constraints of Eq. (4.16), the complete equations of the internal dynamics do not always result in a set of DAE of index 3 as it was shown previously in Chapter 2. In the present method, the equation in DAE form is directly solved without index reduction using the generalized-α scheme [23] summarized in the previous chapter.

4.2.1 DAE index and linear analysis

The discussions presented in Chapter 2 about the differential index and linear stability analysis are adapted for the present formulation.

Let us assume that there is no redundant kinematic constraints in the model i.e. $\mathbf{B}$ is full rank. The matrix defined in Eq. (2.7) can now be redefined as

$$\begin{bmatrix} \mathbf{M} & \mathbf{B}^T & -\mathbf{A} \\ \mathbf{B} & \mathbf{0} & \mathbf{0} \\ \mathbf{D} & \mathbf{0} & \mathbf{0} \end{bmatrix} \qquad (4.17)$$

If this matrix is non-singular, the internal dynamics is represented by an index 3 DAE. If this matrix is singular, the index is higher than 3.

To perform the linear stability analysis, Eqs. (4.14-4.16) are now linearized around an equilibrium point with some increments $\Delta\mathbf{h}$, $\Delta\mathbf{v}$, $\Delta\dot{\mathbf{v}}$, $\Delta\boldsymbol{\lambda}$ and $\Delta\mathbf{u}$. As in Eq. (2.9), this linearized system can be written in the descriptor state space form with

$$\mathbf{M}^\star = \begin{bmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{M} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} \quad ; \quad \mathbf{F}^\star = \begin{bmatrix} \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} \\ -\mathbf{K} & -\mathbf{C} & -\mathbf{B}^T & \mathbf{A} \\ \mathbf{B} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{D} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} \tag{4.18}$$

where $\mathbf{D}$ is the linearization of the servo constraints for a finite variation $\Delta\mathbf{h}$ of the configurations.

In the nonlinear planar example of Fig. 4.1, when the output of the system is fixed, the internal dynamics can be described by the states $(\eta, \dot{\eta})$ of the flexible passive hinge. Analyzing the generalized eigenvalues problem of the linearized system, one finds two eigenvalues (Fig. 4.2): a stable one with a negative real part, located at around -25 rad/s (or 4 Hz), and an unstable one with a positive real part, located at around 28 rad/s (or 4.5 Hz). Hence one pole of the inverse dynamics transfer function is unstable and the system is said to be non-minimum phase. Consequently, integrating the inverse dynamics from an initial state would lead to unbounded inputs and states. Alternatively, stable inversion methods can be used. One can also note that the index of the DAE is 3 and the relative degree for each input/output pairs is 2 in this case (as shown in [98]).

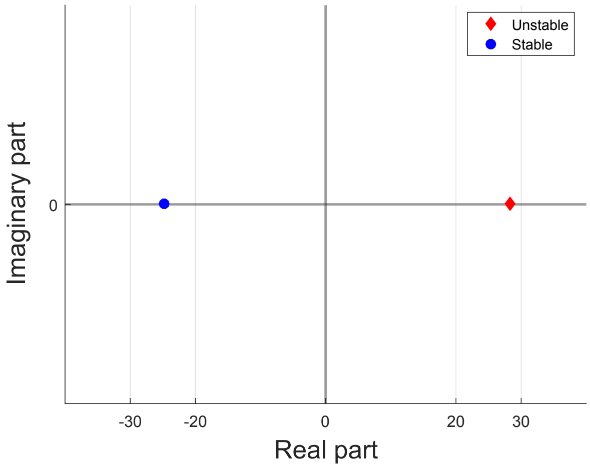


Figure 4.2: Poles of the linearized planar under-actuated cart system.

4.3 Optimization based formulation

Amongst stable inversion methods, the optimization based formulation states the inverse dynamics problem as a constrained optimization problem where a measure of the amplitude of the internal dynamics over the trajectory is minimized [4, 5, 6]. Considering that such a measure can be represented based on an instantaneous function $V(\mathbf{H}(t))$ depending on the nodal configurations, the constrained optimization problem is formulated as the minimization of the objective function J, defined as

$$J = \frac{1}{T}\int_{t_i}^{t_f} V(\mathbf{H}(t))dt \tag{4.19}$$

with the time lap $T = t_f - t_i$, subjected to the internal dynamics equations, denoted by $\mathbf{c}$, defined by Eqs. (4.14-4.16) for $t \in [t_i, t_f]$.

As mentioned in the introduction, one may observe that no initial and final values of $\mathbf{H}$ and $\mathbf{v}$ are needed in this formulation. They are determined by the optimization algorithm itself. In what follows the details of this formulation combined with the FEM on $SE(3)$ are exposed.

4.4 Direct transcription method

Continuous optimization problems can usually be solved using either direct or indirect methods. Here a direct method is used to solve the optimization problem numerically. More particularly, the optimization is carried out using a direct transcription method [18] i.e., the time interval is first discretized in s time steps so that the optimal control problem is reformulated as a discrete *non-linear program* (NLP).

4.4.1 Time discretization

Once the continuous problem is discretized into s time steps of size h, the objective function to be minimized, previously given by Eq. (4.19), can be written as

$$J = \frac{1}{T}\sum_{k=1}^{s} V(\mathbf{H}^k)h \tag{4.20}$$

and the discrete constraints at each time step k are

$$\begin{aligned} \mathbf{M}^k\dot{\mathbf{v}}^k + \mathbf{g}(\mathbf{H}^k, \mathbf{v}^k) + \mathbf{B}^{k,T}\boldsymbol{\lambda}^k - \mathbf{A}\mathbf{u}^k &= \mathbf{0} & (4.21)\\ \boldsymbol{\Phi}(\mathbf{H}^k) &= \mathbf{0} & (4.22)\\ \boldsymbol{\Psi}(\mathbf{H}^k) &= \mathbf{0} & (4.23) \end{aligned}$$

Additional time integration constraints are required to connect the discrete nodal configurations, velocities and accelerations. The Lie group generalized-α time integration scheme [23] presented previously in Eqs. (3.72-3.74) is used. For each time step k,

$$\begin{aligned}
\mathbf{Q}^k - h\mathbf{v}^k - (\frac{1}{2} - \beta)h^2\mathbf{a}^k - \beta h^2\mathbf{a}^{k+1} &= \mathbf{0} && (4.24)\\
\mathbf{v}^{k+1} - \mathbf{v}^k - (1-\gamma)h\mathbf{a}^k - \gamma h\mathbf{a}^{k+1} &= \mathbf{0} && (4.25)\\
(1-\alpha_m)\mathbf{a}^{k+1} + \alpha_m\mathbf{a}^k - (1-\alpha_f)\dot{\mathbf{v}}^{k+1} - \alpha_f\dot{\mathbf{v}}^k &= \mathbf{0} && (4.26)
\end{aligned}$$

As in Eq. (3.69), the vector $\mathbf{Q}^{k,T} = \left[\mathbf{Q}_1^{k,T}, \cdot\!\cdot, \mathbf{Q}_N^{k,T}, \mathbf{Q}_{J,1}^{k,T}, \cdots, \mathbf{Q}_{J,n}^{k,T}\right]$ represents the incremental motion of each node (and joint) I between two consecutive time steps k and $k+1$ and is related to the configuration of each node (and joint) using

$$\mathbf{H}_I^{k+1} = \mathbf{H}_I^k \exp_{SE(3)}(\widetilde{\mathbf{Q}_I^k}) \tag{4.27}$$

After discretization, the unknown variables of the optimization problem are

$$(\mathbf{H}^1, \mathbf{v}^1, \dot{\mathbf{v}}^1, \mathbf{a}^1, \boldsymbol{\lambda}^1, \mathbf{u}^1, \cdots, \mathbf{H}^s, \mathbf{v}^s, \dot{\mathbf{v}}^s, \mathbf{a}^s, \boldsymbol{\lambda}^s, \mathbf{u}^s)$$

with $\mathbf{H}^k = \text{diag}(\mathbf{H}_1^k, \cdots, \mathbf{H}_N^k, \mathbf{H}_{J,1}^k, \cdots, \mathbf{H}_{J,n}^k)$ and each $\mathbf{H}_I^k$ belongs to $SE(3)$, a nonlinear matrix group. In order to solve this problem using classical techniques, meaning techniques that deal with variables defined on a linear space, a reformulation on the Lie algebra $se(3)$ based on incremental configuration variables is proposed.

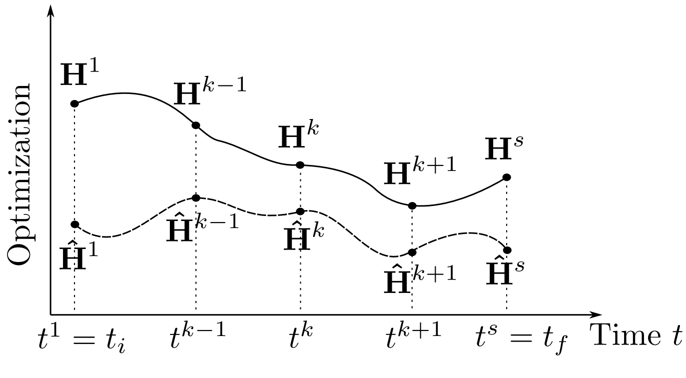


Figure 4.3: Direct transcription method: optimization starting from the initial guess $(\hat{\mathbf{H}}^1, \cdots, \hat{\mathbf{H}}^s)$ that leads to the optimal trajectory $(\mathbf{H}^1, \cdots, \mathbf{H}^s)$.

4.4.2 Incremental formulation of the NLP

In order to formulate the problem in the Lie algebra, an additional assumption is introduced. For that purpose, an equivalent rigid system is first defined by replacing every compliant body by a rigid body with the same total mass, rotational inertia and geometry as in the undeformed configuration. It is assumed that the rigid system is fully actuated and that, when the output trajectory is specified, the continuous rigid trajectory, denoted as $\hat{\mathbf{H}}(t)$, is completely determined. The internal dynamics of the flexible system is thus interpreted as a purely elastic deviation with respect to the rigid trajectory. The latter can therefore serve as the initial guess of the optimization process.

As illustrated in Fig. 4.3, it is proposed to represent the trajectory in terms of increments with respect to the rigid trajectory. One introduces the vector of incremental variables $\mathbf{q}^T = \left[\mathbf{q}_1^T, \cdots, \mathbf{q}_N^T, \mathbf{q}_{J,1}^T, \cdots, \mathbf{q}_{J,n}^T\right]$ which determines the change between the rigid trajectory $\hat{\mathbf{H}}$ and the value $\mathbf{H}$ at the current iteration of the optimization process. At time step k, the relation between the configuration $\mathbf{H}^k$ and the incremental variables $\mathbf{q}^{k,T}$ is

$$\mathbf{H}_I^k = \hat{\mathbf{H}}_I^k \exp_{SE(3)}(\widetilde{\mathbf{q}_I^k}) \tag{4.28}$$

where $\hat{\mathbf{H}}_I^k$ represents the discrete configurations at time step k for the initial guess. As one can see, each incremental vector $\mathbf{q}_I$ represents a variable $\widetilde{\mathbf{q}}_I$ that belongs to the Lie algebra $se(3)$.

It is important to notice the difference between Eq. (4.27) and Eq. (4.28). The former relates the nodal configuration of the system at the same iteration but at two consecutive time steps, i.e. $\mathbf{H}^k$ and $\mathbf{H}^{k+1}$. The latter relates the nodal configuration of the system before and after optimization at a given time step, i.e. $\hat{\mathbf{H}}^k$ and $\mathbf{H}^k$. The relation between the relevant variables is illustrated in Fig. 4.4. Each arrow represents an exponential mapping $\exp_{SE(3)}(\bullet)$ with either time incremental or iteration incremental arguments, i.e. $\mathbf{Q}^k$ and $\mathbf{q}^k$ respectively. The actual design variables $\mathbf{x}$ are thus

$$\mathbf{x} = (\mathbf{q}^1, \mathbf{v}^1, \dot{\mathbf{v}}^1, \mathbf{a}^1, \boldsymbol{\lambda}^1, \mathbf{u}^1, \cdots, \mathbf{q}^s, \mathbf{v}^s, \dot{\mathbf{v}}^s, \mathbf{a}^s, \boldsymbol{\lambda}^s, \mathbf{u}^s)$$

The optimization problem has now vectorial design variables and can be solved using a classical NLP algorithm. As the initial guess is the rigid trajectory, the velocities, accelerations, Lagrange multipliers and control inputs values are initialized with the rigid trajectory values. Initially, there is no elastic deformation in the system, therefore the initial guess for the incremental variables $\mathbf{q}$ is zero.

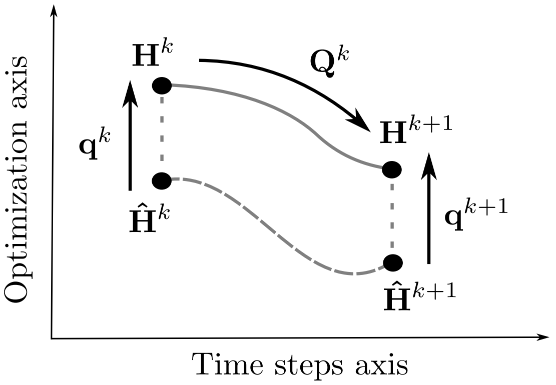


Figure 4.4: Relation between variables $\mathbf{q}^k$, $\mathbf{Q}^k$ and $\mathbf{H}^k$.

4.4.3 Objective function

Since the motion in the flexible structure is considered as a purely elastic motion about the rigid configurations, the function $V(\mathbf{H}^k)$ in the discrete settings (see Eq. (4.20)) can be defined either as the strain energy of the system or based on the strain measures of the finite element model. For example, if the system has a flexible joint with a relative coordinate α_J and a stiffness $k^\star$ (as defined in Section 3.4.1), $V(\mathbf{H}^k)$ can simply be defined as

$$V(\mathbf{H}^k) = \frac{1}{2}(\alpha_J^k - \alpha_J^0)^2 \; k^\star \tag{4.29}$$

and the gradient of this objective function would be

$$\frac{\partial J}{\partial \alpha_J} = \frac{1}{T}(\alpha_J^k - \alpha_J^0) \; k^\star \; h \tag{4.30}$$

If a flexible link system is considered and flexibility in the system is modeled thanks to beam elements, $V(\mathbf{H}^k)$ can simply be defined as

$$V(\mathbf{H}^k) = \frac{1}{2}\sum_{j=1}^{n_{el}} L_j \; \boldsymbol{\epsilon}_j^{k,T}(\mathbf{H}^k)\mathbf{K}_j\boldsymbol{\epsilon}_j^k(\mathbf{H}^k) \tag{4.31}$$

where n_{el} is the number of beam elements, L_j is the length of the beam element j, $\mathbf{K}_j$ is the beam element stiffness matrix and the strains $\boldsymbol{\epsilon}_j^k$ are defined as in (3.50).

The gradient of the objective function in this case is

$$\frac{\partial J}{\partial \mathbf{q}^k} = \frac{1}{T}\left[\sum_{j=1}^{n_{el}} L_j \ \boldsymbol{\epsilon}_j^{k,T} \mathbf{K}_j \frac{\partial \boldsymbol{\epsilon}_j^k}{\partial \mathbf{q}^k}\right] h \tag{4.32}$$

and the coupling terms in $\partial \boldsymbol{\epsilon}_j / \partial \mathbf{q}$ are limited to the pair of nodes A and B connected by beam elements, which enforces a sparse pattern of the gradient. The derivative of the strains with respect to the optimization variable $\mathbf{q}$ is given by

$$\frac{\partial \boldsymbol{\epsilon}_j^k}{\partial \mathbf{q}^k} = \begin{bmatrix} -\mathbf{T}^{-1}(-\mathbf{d}^k)\mathbf{T}(\mathbf{q}_A^k) & \mathbf{T}^{-1}(\mathbf{d}^k)\mathbf{T}(\mathbf{q}_B^k) \end{bmatrix} \tag{4.33}$$

where the operators $\mathbf{T}^{-1}(\bullet)$ and $\mathbf{T}(\bullet)$ are as defined in Eq. (3.11) and result from the differentiation of the logarithmic map $\log_{SE(3)}(\bullet)$ in Eq. (3.48) and the exponential map $\exp_{SE(3)}(\bullet)$ in Eq. (4.28).

4.4.4 Constraints gradient

Regarding the optimization constraints $\mathbf{c}$, they will be denoted as $\mathbf{c}_m$ and $\mathbf{c}_\alpha$ for the equation of motion (4.21-4.23) and the time integration equations (4.24-4.26) respectively. At each time step k, these constraints only involve a subset $\mathbf{x}^k = (\mathbf{q}^k, \mathbf{v}^k, \dot{\mathbf{v}}^k, \mathbf{a}^k, \boldsymbol{\lambda}^k, \mathbf{u}^k)$ of the design variables $\mathbf{x}$. For the equations of motion $\mathbf{c}_m$, the gradient is

$$\frac{\partial \mathbf{c}_m^k}{\partial \mathbf{x}^k} = \begin{bmatrix} \mathbf{K}^k\mathbf{T}^\star(\mathbf{q}^k) & \mathbf{C}^k & \mathbf{M}^k & \mathbf{0} & \mathbf{B}^{k,T} & -\mathbf{A} \\ \mathbf{B}^k\mathbf{T}^\star(\mathbf{q}^k) & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{D}^k\mathbf{T}^\star(\mathbf{q}^k) & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} \tag{4.34}$$

where $\mathbf{K}^k$ and $\mathbf{C}^k$ are the tangent stiffness and damping matrix respectively and $\mathbf{D}^k$ is the gradient of the servo constraints. For the time integration equations $\mathbf{c}_\alpha$, the gradient is

$$\frac{\partial \mathbf{c}_\alpha^k}{\partial \mathbf{x}^k} = \begin{bmatrix} \mathbf{T}^\star(\mathbf{Q}^k) & -h\mathbf{I} & \mathbf{0} & -(\frac{1}{2}-\beta)h^2\mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -\mathbf{I} & \mathbf{0} & -(1-\gamma)h\mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & -\alpha_f\mathbf{I} & \alpha_m\mathbf{I} & \mathbf{0} & \mathbf{0} \end{bmatrix} \tag{4.35}$$

and

$$\frac{\partial \mathbf{c}_\alpha^k}{\partial \mathbf{x}^{k+1}} = \begin{bmatrix} \mathbf{T}^\star(\mathbf{Q}^{k+1}) & \mathbf{0} & \mathbf{0} & -\beta h^2\mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & \mathbf{0} & -\gamma h\mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & -(1-\alpha_f)\mathbf{I} & (1-\alpha_m)\mathbf{I} & \mathbf{0} & \mathbf{0} \end{bmatrix} \tag{4.36}$$

where $\mathbf{I}$ is the identity matrix and the matrix $\mathbf{T}^\star$ is a block diagonal matrix containing the tangent operator $\mathbf{T}$ from Eq. (3.11) applied to each incremental configurations $\mathbf{q}_I^k$ (or $\mathbf{Q}_I^k$). Finally, the resulting constraint gradient matrix $\partial\mathbf{c}/\partial\mathbf{x}$ is made up of these matrices at each time step, stacked on each other to build a block diagonal sparse matrix

$$\frac{\partial\mathbf{c}}{\partial\mathbf{x}} = \begin{bmatrix} \frac{\partial\mathbf{c}_m^1}{\partial\mathbf{x}^1} & \mathbf{0} & & \dots & \mathbf{0} \\ \frac{\partial\mathbf{c}_\alpha^1}{\partial\mathbf{x}^1} & \frac{\partial\mathbf{c}_\alpha^1}{\partial\mathbf{x}^2} & & & \\ \mathbf{0} & \frac{\partial\mathbf{c}_m^2}{\partial\mathbf{x}^2} & & & \vdots \\ & \frac{\partial\mathbf{c}_\alpha^2}{\partial\mathbf{x}^2} & \frac{\partial\mathbf{c}_\alpha^2}{\partial\mathbf{x}^3} & & \\ & & \ddots & & \\ \vdots & & & \frac{\partial\mathbf{c}_m^{s-1}}{\partial\mathbf{x}^{s-1}} & \mathbf{0} \\ & & & \frac{\partial\mathbf{c}_\alpha^{s-1}}{\partial\mathbf{x}^{s-1}} & \frac{\partial\mathbf{c}_\alpha^{s-1}}{\partial\mathbf{x}^{s}} \\ \mathbf{0} & & \dots & \mathbf{0} & \frac{\partial\mathbf{c}_m^{s}}{\partial\mathbf{x}^{s}} \end{bmatrix} \tag{4.37}$$

It is important to note that, although the tangent stiffness $\mathbf{K}^k$ matrices and tangent damping $\mathbf{C}^k$ matrices are non-linear and should be re-evaluated for the constraint gradient at each iteration, the optimization process may still converge if they are kept constant over the iterations. Indeed, thanks to the local representation coming from the use of the $SE(3)$ formalism, these matrices do not vary much through the optimization process. Hence, for simple systems, this assumption can lead to improved computational time. However, in the more complex systems studied here, using constant $\mathbf{K}^k$ and $\mathbf{C}^k$ in the constraint gradient leads to more iterations of the optimization and an increased computational time. In the following examples, the whole constraint gradient is thus computed at every iteration.

4.4.5 Optimization algorithm

The combination of the FEM formalism and the direct transcription method leads to a large optimization problem with a high number of design variables and constraints. However, the problem is sparse as coupling terms in the discrete constraints are limited to consecutive time steps (see Eq. (4.24-4.26)) and nodes directly connected in the finite element mesh. Some non-linear optimization solvers, such as KNITRO, IPOPT and FMINCON, were tested with appropriate options and algorithms to deal with large scale and sparse problems. Eventually, the *interior point* algorithm in the FMINCON solver of Matlab® seemed to perform more efficiently for the present test cases and was used to solve the inverse dynamics problem. The complete inverse dynamics formulation is integrated in the GECOS development code. The tolerance for the satisfaction of the constraints of the optimization problem were fixed to $tol = 10^{-6}$ in order to guarantee that both the servo-constraints and the holonomic constraints are satisfied with a sufficient level of accuracy by the numerical solution.

For computational efficiency, the gradients of the objective function and the constraint equations, computed previously, are provided to the solver. The Hessian matrix is computed numerically by the solver using the *lbfgs* quasi-Newton approximation.

The optimization algorithm is summarized in Fig. 4.5. Starting from the initial guess, the optimizer determines an incremental step for the design variables in order to improve the objective function and the constraints. The objective function and constraints are then evaluated again to check the convergence of the process. Eventually a bounded solution is found for the flexible system. The present method is not intended to run in real-time, therefore, in the coming discussions, all the inverse dynamics solutions have been computed prior their use as feedforward input on simulated and experimental manipulators.

4.5 2D under-actuated cart example

The above formulation of the inverse dynamics is now applied to the illustrative example shown in Fig. 4.1. The parameters of the system are given in Table 4.1. The inertias $I_{1,2,3}$ of each body are considered equal in all directions. The smooth linear trajectory the cart has to follow goes from $x = -1$ m to $x = 1$ m. At the same time, the end-effector must follow a circular arc trajectory with a radius of 1 m from $(x_{\text{eff}}, y_{\text{eff}}) = (-1, -1.5)$ to $(x_{\text{eff}}, y_{\text{eff}}) = (1, -1.5)$ in 1.5 s. As shown in Fig. 4.2, since the system is non-minimum phase some pre-actuation and post-actuation of the system can be expected. Therefore, the simulation time is set to last 2 s.

Cart	$m_c = 3$ kg		
Link 1	$m_1 = 6.875$ kg	$I_1 = 0.5743$ kg.m^2	$l_1 = 1$ m
Link 2,3	$m_{2,3} = 3.4375$ kg	$I_{2,3} = 0.0723$ kg.m^2	$l_{2,3} = 0.5$ m
	$k = 50$ N.m/rad	$d = 0.25$ N.m.s/rad	

Table 4.1: Parameters of the planar under-actuated cart system.

In the objective function of the optimization, the functional V is defined in the discrete settings as presented in Eq. (4.29) with $\alpha_J = \eta$ and the gradient of the objective function is defined as in Eq. (4.30).

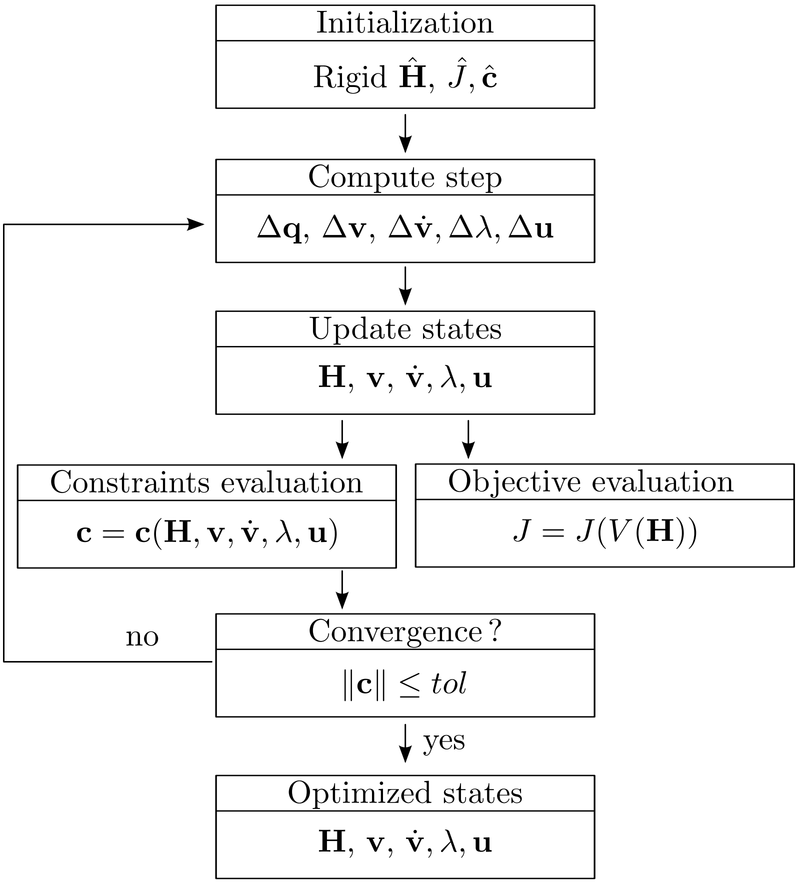


Figure 4.5: Schematic graph of the optimization algorithm used to solve the inverse dynamic problem.

The numerical parameters are fixed as in [4]: the time step size of the simulation is set to 100 Hz and the time frame is discretized into $s = 200$ time steps. Regarding the numerical parameters, the spectral radius ρ_∞ of the generalized-α method is set to 0.3 ($\beta = 0.59$, $\gamma = 1.04$, $\alpha_m = -0.31$ and $\alpha_f = 0.23$). The initial guess is the motion of the equivalent rigid system, i.e. a system where the passive hinge joint is locked in its initial position.

4.5.1 Results and discussion

With the tolerance $tol = 10^{-6}$, the optimization process is completed after 5 iterations and lasts about 2 minutes (using a x64 bits i7-4600u CPU with 16 Gb RAM memory). The convergence of the coordinate η is shown in Fig. 4.6. The vertical black lines show when the initial and final times of the output trajectory. One can

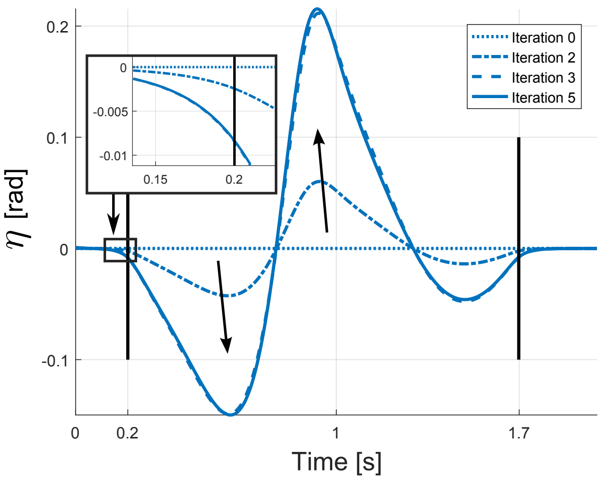


Figure 4.6: Convergence process of the passive hinge coordinate η of the under-actuated cart system.

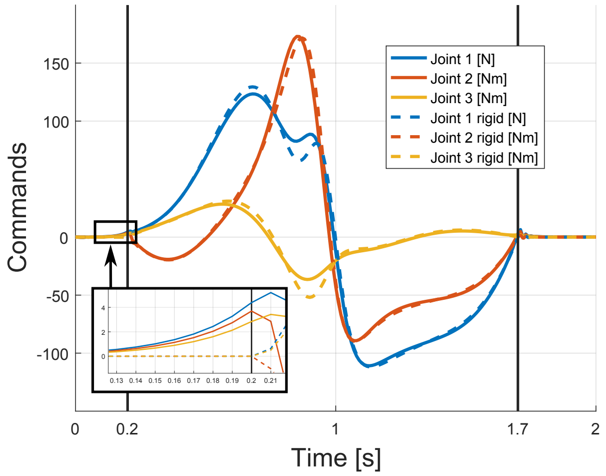


Figure 4.7: Reference input commands $\mathbf{u}$ of an under-actuated cart system.

see that at the initial guess (Iteration 0), the flexible joint is indeed locked and no internal motion of the arm is present. After the third iteration, the optimization process is already rather close to the final solution (Iteration 5). One also observes that the flexible joint η starts to move before (after) the cart and the end-effector start (finish) tracking the trajectory. Some pre- and post- actuation in the system is therefore present. This can be observed in the reference input commands $\mathbf{u}$ shown in Fig. 4.7. The delay between actuation and output motion is a consequence of the non-minimum phase behavior of the system.

In order to verify the flexible reference inputs, they were applied to the system and a direct dynamic analysis was performed. The resulting trajectory of the end-effector is shown in Fig. 4.8: compared to a control with the rigid reference $\mathbf{u}_{\text{rigid}}$, the flexible reference inputs $\mathbf{u}$ allow one to reduce the error on the tracked trajectory. The relative error at each time step k is calculated using

$$e^k = \frac{\|\mathbf{y}_{\text{presc}} - \mathbf{y}\|}{\|\mathbf{y}_{\text{presc}}\|} \tag{4.38}$$

where $\|\bullet\|$ is the classical Euclidean norm or L_2 norm. The relative rms error is then calculated as

$$e_{rms} = \sqrt{\frac{1}{s}\sum_{1}^{s}(e^k)^2} \tag{4.39}$$

When the flexible reference inputs are used, $e_{rms} = 2 \times 10^{-3}$ whereas it grows to $e_{rms} = 3.2 \times 10^{-2}$ when the rigid reference inputs are considered. One can note that in the former case, the remaining error is only caused by finite numerical tolerances that are used by the algorithm. For example, if the tolerance is reduced to $tol = 10^{-9}$, this error becomes $e_{rms} = 5 \times 10^{-5}$.

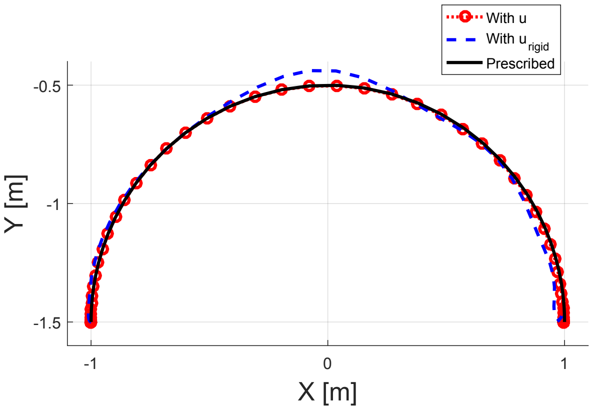


Figure 4.8: End-effector trajectory of the under-actuated cart system using inputs $\mathbf{u}$ and $\mathbf{u}_{rigid}$.

In Fig. 4.9 the phase plot of the internal dynamics $(\eta, \dot{\eta})$ is shown. In this figure, the stable manifold $\boldsymbol{\zeta}_{\text{s}}$ and the unstable manifold $\boldsymbol{\zeta}_{\text{us}}$ are also represented. Following the theoretical results of the stable inversion method [40, 100, 22], the computed inputs $\mathbf{u}$ indeed drive the internal dynamics along the unstable manifold initially. This dynamics then follows the stable manifold until reaching the end of the trajectory.

Note As pointed out in the introduction (Section 1.3), direct transcription methods are more robust and can still converge although a poor initial guess is given. Indeed, for simple flexible systems such as this planar under-actuated cart, the op-

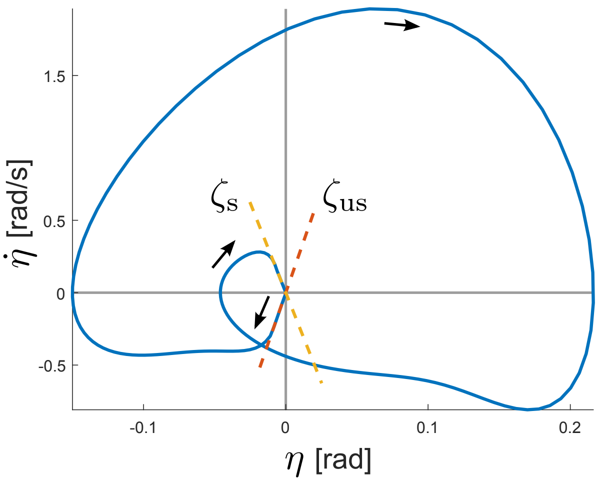


Figure 4.9: Phase plot of the internal dynamics η.

timization process is able to converge to a sensible solution even if all the design variables of the initial guess are zeros. As can be expected though, more iterations are needed in such cases. A similar observation was done for a one actuated dof flexible beam system. For more complex test cases, however, the optimization process is not able to converge in a reasonable amount of iterations. Therefore, in the following spatial examples, it is important to provide sensible initial guesses to the solver.

4.6 Summary

In this chapter, the proposed method to generate a feedforward control input for general flexible manipulators is presented. First, the servo constraints prescribing the trajectory of the output are reformulated in the $SE(3)$ matrix Lie group and the inverse dynamics problem is formulated as a set of DAE on a Lie group. Next, the stable inversion method is formulated as a constrained optimization problem formulated on a matrix Lie group. The objective function is defined to give a measure of the amplitude of the internal dynamics of the flexible system. The constraints are given by the equations of the inverse dynamics. The complete optimization problem is then discretized using a direct transcription method leading to a non-linear program with non-linear design variables defined on $SE(3)$. In order to use classical optimization tools defined on a linear space, it is proposed to reformulate the design variables as incremental variables which belong to the Lie algebra $se(3)$. The method, implemented in the GECOS software is illustrated with a planar under-actuated system from the literature.

Chapter 5

Simulation results

The geometric model and inverse dynamics formulations presented in the previous chapters are now applied for some simulated spatial flexible manipulators. The present chapter demonstrates how the proposed method is able to solve the inverse dynamics problem of systems that have various kinematic topologies and various internal dynamics characteristics. First, manipulators with a serial kinematic topology are considered. When varying the parameters of the systems, the solution of the inverse dynamics problem can be more or less affected by the non-minimum phase characteristic of the internal dynamics. Then, the application of the present method to parallel manipulators is discussed. In these cases, the location of the poles of the inverse dynamics transfer function highly depends on the modeling assumptions of the links and the parameters of the system.

In the coming simulation examples, all end-effector trajectories are parameterized using a trajectory profile $p(t)$ defined as a seventh order polynomial (as shown in Fig. 5.1). In this way, the smoothness of the position, velocity, and acceleration is assured from the beginning t_i to the end t_f of the trajectory. The starting value $p(t_i)$ and ending value $p(t_f)$ is case dependent.

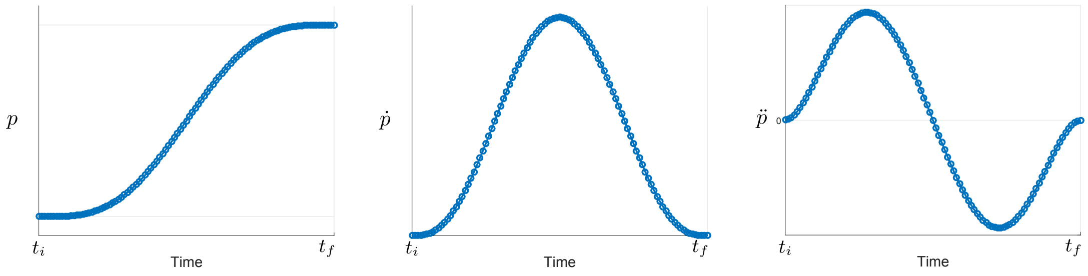


Figure 5.1: Trajectory $p(t)$ as a seventh order polynomial.

5.1 3D serial arm system

The serial 3 dof manipulator first illustrated in Section 3.6.1 and recalled in Fig. 5.2 is now considered [67]. Serial mechanisms are the most widely used topology in robotic manipulators: a simple chain of bodies connects the end-effector to the base of the manipulator.

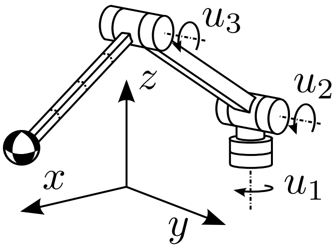


Figure 5.2: Serial 3D arm system with one rigid body and 4 beam elements.

5.1.1 Description of the problem

The system is composed of two links: the first one connects the system to the base and the second connects the end-effector to the rest of the system. The end-effector is modeled as a point mass m_{end} at the tip of the system. Both links a have length l and a tubular square cross section. The first link has a side length a_1 and an edge thickness e_1. The second link has a side length a_2 and an edge thickness e_2. While the former has a greater cross section, i.e. $a_1 > a_2$, and is considered as a rigid body element, the second link is considered flexible and is modeled using 4 beam elements. The description of the FEM beam formulation on $SE(3)$ is summarized in Section 3.5.2. The first link is connected to the base thanks to the first two hinge joints, controlled using input torques u_1 and u_2. The second link is connected to the upper arm through the third hinge joint, controlled using input torque u_3. The outputs of the system are the x, y and z components of the end-effector position $\mathbf{y}_{\text{eff}}$ and $r = 3$. The first hinge joint has its axis along axis z. The second and the third hinge joints initially have their axis along axis y. In the initial position, each link makes a $\pi/4$ angle with respect to the x axis. The complete model is composed of 10 nodes, 2 rigid body elements, 4 beam elements and 3 kinematic joint elements as previously shown in Fig. 3.11(b).

The trajectory $\mathbf{y}_{\text{presc}}$ the end-effector has to follow is a planar circular arc of radius $l/2$ in the yz plane. The end-effector trajectory is defined using the seventh order polynomial $p(t)$ as

$$\begin{aligned} x &= 2l\cos(\pi/4) = \text{constant} \\ y &= l(1+\sin(p(t)))/2 \\ z &= l\cos(p(t))/2 \end{aligned} \tag{5.1}$$

with $p(t_i) = -\pi/2$ and $p(t_f) = \pi/2$. The trajectory is covered in 1.1 s. The material parameters and dimensions of the 3D flexible arm can be found in Table 5.1. No material damping is considered in this case.

First link	$l = 1$ m	$a_1 = 0.05$ m	$e_1 = 0.01$ m
			$\rho = 2700$ kg/m^3
Second link	$l = 1$ m	$a_2 = 0.0075$ m	$e_2 = 0.0015$ m
	$E = 70$ GPa	$\nu = 0.3$	$\rho = 2700$ kg/m^3
End-effector		$m_{end} = 0.1$ kg	

Table 5.1: Parameters of the serial arm system.

The numerical parameters of the model are set to capture the dynamics of the system with a bandwidth of 100 Hz: in the initial position, the first seven modes of the flexible structure are within this frequency. As shown in Fig. 5.3, several poles of the linearized inverse dynamics problem are also within this bandwidth; the first unstable pole is located at 13 Hz. Since there are some unstable poles, the system is non-minimum phase. Computing the matrix of Eq. (4.17), one finds that the index of the system is 3. The system is discretized into $s = 300$ steps, leading to a time step size h of 0.005 s (200 Hz), which is twice the bandwidth and more than ten times the first unstable frequency. For the generalized-α method, a spectral radius of $\rho_\infty = 0$ is considered ($\beta = 1$, $\gamma = 1.5$, $\alpha_m = -1$ and $\alpha_f = 0$) to eliminate higher frequency content. Since flexibility is described using beam elements, the objective function and its gradient are defined in the discrete settings as in Eq. (4.31) and Eq. (4.32).

5.1.2 Results and discussion

As previously, a complete rigid system is considered as initial guess of the optimization. The beams of the second link are replaced with a rigid body with the same geometrical and material properties. The resulting control input is denoted as the rigid reference input $\mathbf{u}_{\text{rigid}}$. With the tolerance $tol = 10^{-6}$, one can see from Fig. 5.4

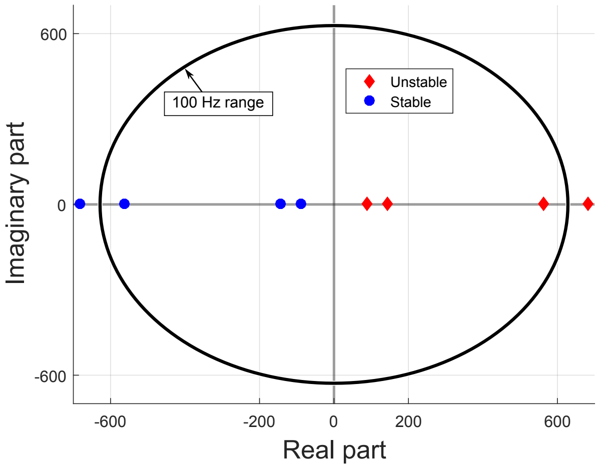


Figure 5.3: Poles of the linearized spatial serial arm.

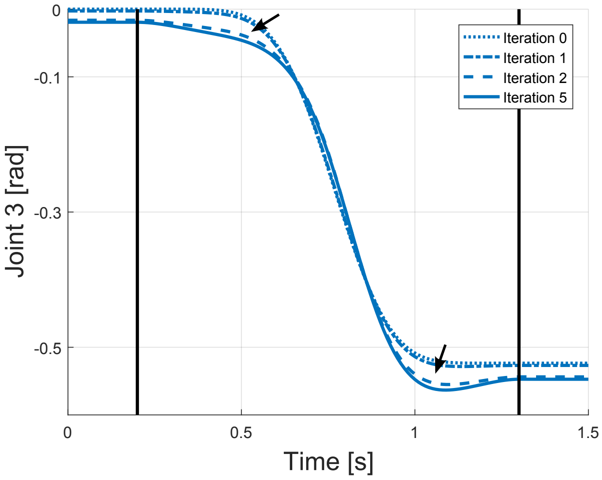


Figure 5.4: Convergence of the iteration process for the third joint of the serial arm.

that the rigid initial guess, corresponding to *Iteration 0*, converges to the optimized solution after 5 iterations. Using a x64 bits i7-4600u CPU with 16 Gb RAM memory, this process lasts 186 seconds. The command input $\mathbf{u}$ from the optimized solution is called the flexible reference input. The reference inputs $\mathbf{u}$ and $\mathbf{u}_{\text{rigid}}$ are compared in Fig. 5.5. Although the flexible system is non-minimum phase and some visible differences are observed, pre- and post-actuation in the input commands are of the order of 10^{-4}Nm and are hardly visible.

In order to verify the flexible reference input $\mathbf{u}$, both inputs $\mathbf{u}$ and $\mathbf{u}_{\text{rigid}}$ are applied to the flexible system and a direct dynamic analysis is performed. These inputs lead to end-effector trajectories shown in Fig. 5.6. Since the trajectory should be circular, the radius of these resulting trajectories are shown in Fig. 5.7. One can observe that the rigid reference input $\mathbf{u}_{\text{rigid}}$ leads to some oscillation of the

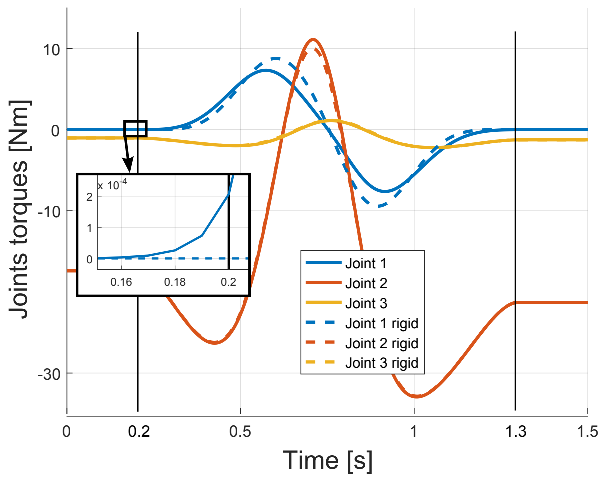


Figure 5.5: Input commands of the flexible 3D serial arm.

end-effector around the nominal circular trajectory. These oscillations are strongly reduced when flexibility is considered. The relative rms error e_{rms} from Eq. (4.39) is equal to 10^{-2} when $\mathbf{u}_{\text{rigid}}$ is used as input and drops down to $e_{rms} = 2 \times 10^{-5}$ when $\mathbf{u}$ is used with $tol = 10^{-6}$. In the latter case, the remaining error is only caused by finite numerical tolerances that are used by the algorithm as it was for the planar under-actuated cart system. For example, if the tolerance is reduced to $tol = 10^{-8}$, this error becomes $e_{rms} = 3.3 \times 10^{-7}$.

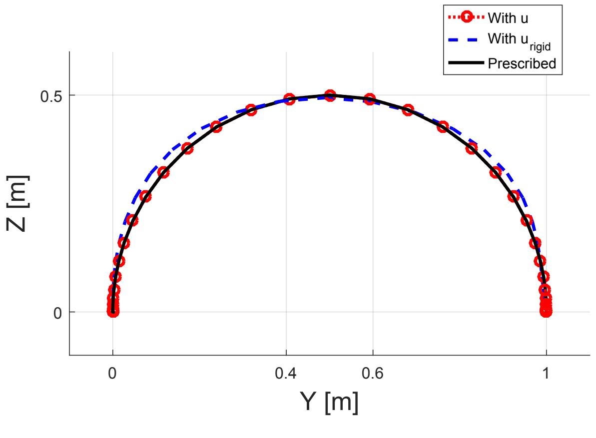


Figure 5.6: End-effector trajectory of the flexible 3D arm using $\mathbf{u}$ and $\mathbf{u}_{\text{rigid}}$.

One can also be interested in the behavior of the internal dynamics of the system. A measure of this internal dynamics can be associated with the deformation $\boldsymbol{\epsilon}$ of the arm. It is also important to note that it is assumed that the internal dynamics is mainly influenced by the lowest pole of the system. Also, the unstable and stable manifolds are spanned by the unstable eigenvectors $\boldsymbol{\zeta}_{\text{us}}$ and the stable eigenvectors $\boldsymbol{\zeta}_{\text{s}}$ respectively. However, these eigenvectors are defined in the state space $(\mathbf{H}, \mathbf{v})$ of the

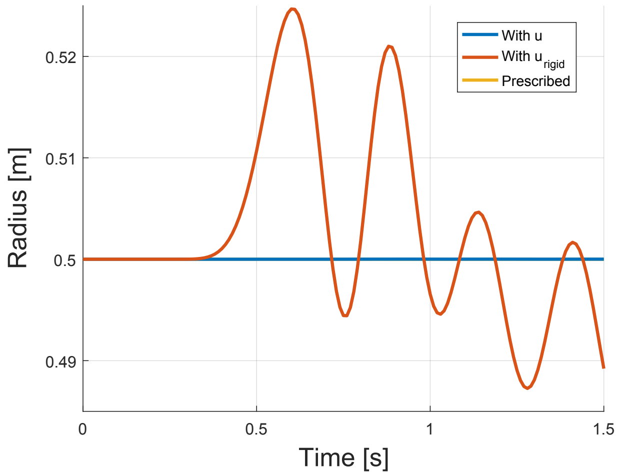


Figure 5.7: Radius of the resulting trajectory compared to the desired trajectory radius of $l/2 = 0.5$m.

system. One can express $\boldsymbol{\zeta}_{\text{us}}$ and $\boldsymbol{\zeta}_{\text{s}}$ in the $(\boldsymbol{\epsilon}, \dot{\boldsymbol{\epsilon}})$ space using Eq. (3.51). In Fig. 5.8, the dynamics of the deformation measure is considered and represented in the $(\boldsymbol{\epsilon}, \dot{\boldsymbol{\epsilon}})$ space for the first beam element of the flexible link. In this figure, the projection of the first stable and unstable eigenvectors on the $(\boldsymbol{\epsilon}, \dot{\boldsymbol{\epsilon}})$ plane are also represented. As expected, it is observed that the dynamics starts in the neighborhood of the first unstable eigenvector and ends in the neighborhood of the first stable eigenvector. The slight pre-actuation is a motion on the unstable manifold on which the system starts. Comparatively, the slight post-actuation is a motion on the stable manifold on which the system ends. From this example, although the system is minimum-phase, the system is rather stiff and the amplitude of the internal dynamics remains limited.

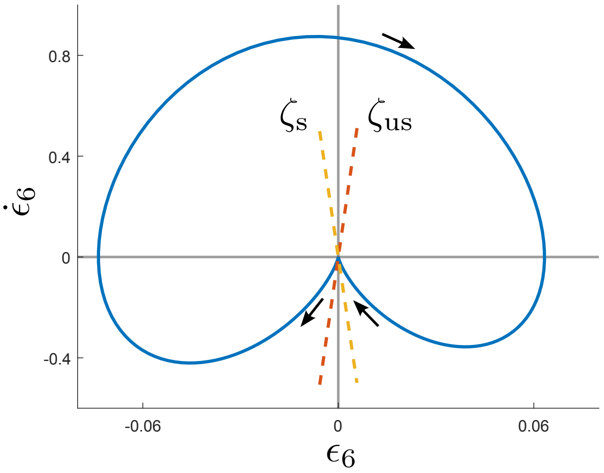


Figure 5.8: Phase plot of the sixth component of the deformation $\boldsymbol{\epsilon}$ (bending around local z axis) in the first beam element of the flexible link.

5.1.3 Alternative architectures

In the first serial example, the system is non-minimum phase and a stable inversion method is needed to solve the inverse dynamics problem. However, the optimized inputs were rather close to the initial guess and very little pre- and post-actuation of the system was needed. Two alternative architectures of the serial manipulator are now considered. Their impact on the internal dynamics of the structure and the computed inverse dynamics solution is investigated.

Flexible first link and rigid second link

Previously, the end-effector was directly connected to the flexible component of the system: the second link. In contrast, in this section, the first link is considered flexible and the second rigid. There is no difference in the way the present solver handles the problem. However, it can be noted that in that case, the resulting inverse dynamics problem has its lowest unstable pole located above the bandwidth set by the time step size (see Fig 5.9). Therefore, from a numerical point of view, the system can directly be integrated forward in time if some numerical dissipation is introduced.

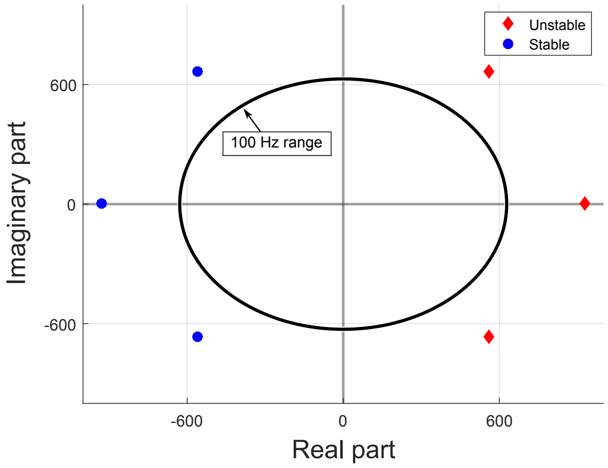


Figure 5.9: Poles of the serial arm with a flexible first link and a second rigid link.

Rigid first link and very flexible second link

The second link is now designed as a beam with a full rectangular cross section with one dimension a_2 much smaller than the other b_2. As a result the bending deformation can be larger around the vertical z axis. This more flexible configuration is shown in Fig. 5.10. The first link and the end-effector mass are kept as in Table 5.1.

The parameters of the more flexible second link are shown in Table 5.2. In this case, some material damping modeled as a Rayleigh damping with coefficients α and β is considered in the beams.

Second link	$l = 1$ m	$a_2 = 0.0015$ m	$b_2 = 0.1$ m
	$E = 70$ GPa	$\nu = 0.3$	$\rho = 2700$ kg/m^3
	$\alpha = 1 \times 10^{-4}$	$\beta = 1 \times 10^{-2}$	

Table 5.2: Parameters of a serial arm system with a more flexible second link.

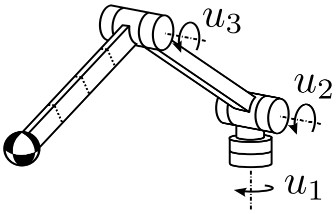


Figure 5.10: Serial 3D arm system with a more flexible second link.

With this alternative serial manipulator, the system still has unstable real poles but one can notice in Fig. 5.11 that they are closer to the imaginary axis. This time, the first pole in the initial configuration is located at 3 Hz.

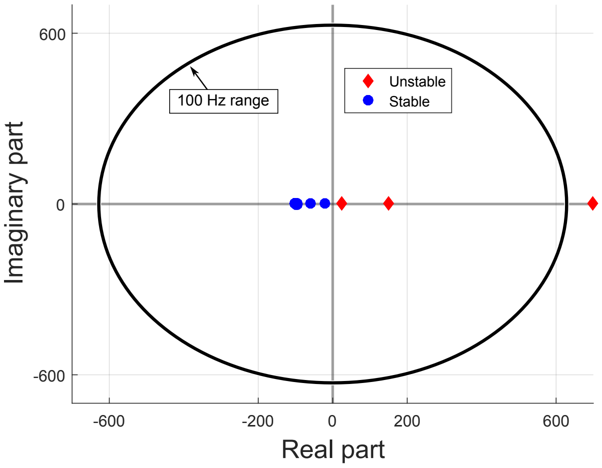


Figure 5.11: Poles of the serial arm with a more flexible second link.

The prescribed trajectory is the same as defined by Eqs. (5.1) but it now has to be tracked in a shorter time of 0.9 s. With the same tolerance *tol*, the optimization process in this case needs 14 iterations and 280 s to converge. The flexible reference inputs $\mathbf{u}$ in this case, shown in Fig. 5.12, vary more from the initial rigid guess. Pre- and post-actuation of the system are now clearly visible. In this case, the unstable behavior of the internal dynamics clearly affects the system's response. One can see from the phase plot in Fig. 5.13, that the bending deformation around the z axis starts on the first unstable eigenvector. Eventually it is driven to end on the first stable eigenvector. The amplitude of the deformation is also larger (ten times) than in the first serial example. With the rigid reference input $\mathbf{u}_{\text{rigid}}$, this simulated very flexible manipulator is not able to track the prescribed trajectory as one can observe in Fig. 5.14. It is important to point out that although some actuation is present in the joints, the end-effector does not actually move during the pre- and post-actuation period.

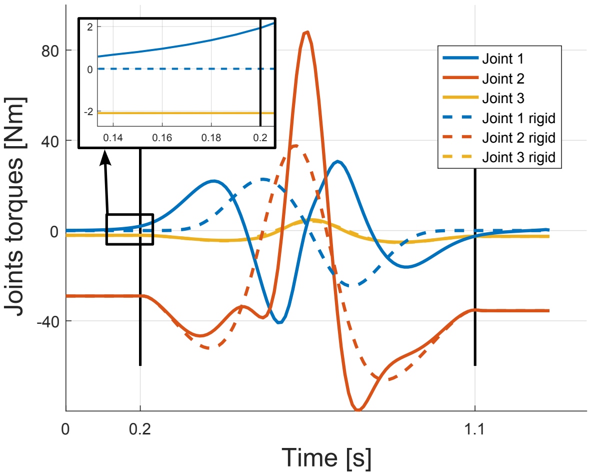


Figure 5.12: Input commands of the serial arm with a more flexible second link.

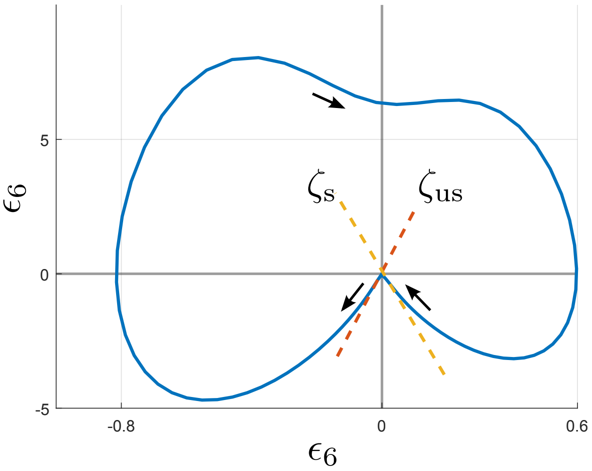


Figure 5.13: Phase plot of the sixth component of the deformation $\boldsymbol{\epsilon}$ (bending around local z axis) in the first beam element of the very flexible link.

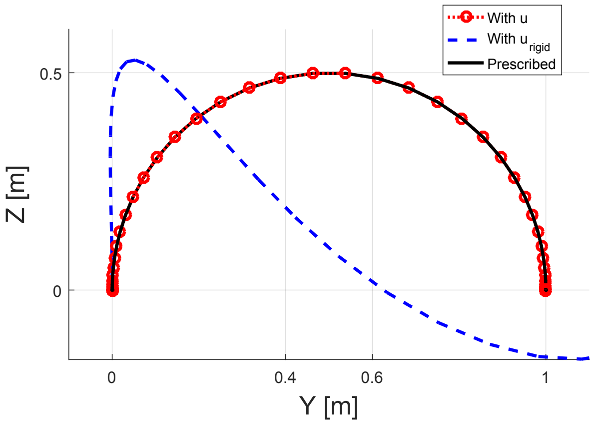


Figure 5.14: End-effector trajectory of the very flexible serial arm using $\mathbf{u}$ and $\mathbf{u}_{\text{rigid}}$.

5.2 3D parallel arm system

Parallel manipulators are structures where a closed chain kinematic exist. The end-effector is connected to the base through several links, leading to a naturally more stiff structure [75].

5.2.1 Description of the problem

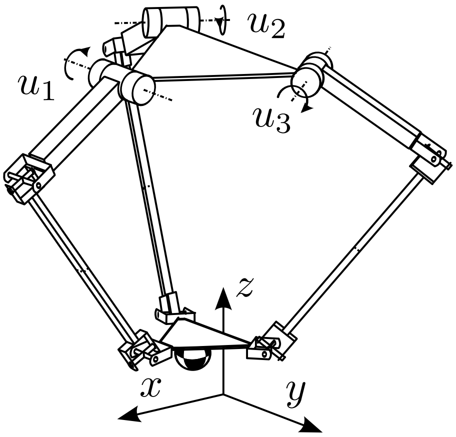


Figure 5.15: Parallel 3D arm system with 3 rigid bodies and 6 beam elements.

The present method is now applied to a manipulator with a closed chain kinematics: a 3 dof Delta robot [32]. It is made of an upper plate and a lower plate, both are equilateral triangles but the upper one has a side length l_{up} whereas the lower one is smaller and has a side length l_{low}. The point mass end-effector is located at the center of the lower plate as shown in Fig. 5.15. Three actuated hinge joints are located at each corner of the upper plate and have their axis parallel to the opposite side of the triangle. They are controlled using the input torques u_1, u_2 and u_3. To control the end-effector, these hinge joints are connected to the lower plate using three arms. Each one is made of two links, an upper link and a lower link, and has two passive universal joints to connect them. The upper links are rigid tubes with a square cross section of dimensions $a_1 \times a_1 \times l_1$ and thickness e_1. The lower links are flexible tubes with dimensions $a_2 \times a_2 \times l_2$ and thickness e_2 and are modeled using 2 beam finite elements. In the initial configuration, the center of gravity of each plate is coincident with the z axis and the three upper arms are horizontal. The initial vertical distance between the upper plate and the end-effector is z_{init}. The total mass of the the lower plate and the end-effector is m_{end}. The trajectory to be followed is a planar circular arc of radius l_{low} in the horizontal xy plane to be covered in 0.6 s. As for the serial example, the motion profile is constructed using a seventh order polynomial $p(t)$:

$$\begin{aligned} x &= l_{low} \cos p(t) \\ y &= l_{low}(1 + \sin p(t)) \\ z &= -z_{init} = \text{constant} \end{aligned} \tag{5.2}$$

with $p(t_i) = -\pi/2$ and $p(t_f) = \pi/2$.

The geometrical and material parameters of the parallel manipulator are given in Table 5.3. Again no material damping is considered in this parallel manipulator case. The bandwidth considered in this simulation is still 100 Hz and at least five eigenmodes of the flexible structure lay within this bandwidth. When considering the poles (in the range of interest) of the inverse dynamics problem at the initial time, one can observe that all are located close to the imaginary axis (see Fig. 5.16) and that the equilibrium point is not hyperbolic. As a result, nothing can be said regarding the stability of the internal dynamics [105]. Since the stable inversion techniques require that the system starts and ends on a hyperbolic equilibrium point, this example may challenge the limits of such techniques. In this case, the index of the DAE is 3. The time step size is set to be 0.005 s. For the generalized-α method, a spectral radius of $\rho_\infty = 0$ is considered ($\beta = 1$, $\gamma = 1.5$, $\alpha_m = -1$ and $\alpha_f = 0$). Since flexibility is modeled with flexible beam elements, the objective function from Eq. (4.31) and its gradient Eq. (4.32) are the same as for the serial manipulator example.

Upper and lower plates	$l_{up} = 0.25$ m	$l_{low} = 0.1$ m	
Upper links	$l_1 = 0.25$ m	$a_1 = 0.05$ m	$e_1 = 0.01$ m
			$\rho = 2700$ kg/m^3
Lower links	$l_2 = 0.3884$ m	$a_2 = 0.0075$ m	$e_1 = 0.0015$ m
	$E = 70$ GPa	$\nu = 0.3$	$\rho = 2700$ kg/m^3
End-effector	$z_{init} = 0.7$ m	$m_{end} = 0.1$ kg	

Table 5.3: Parameters of the parallel arm system.

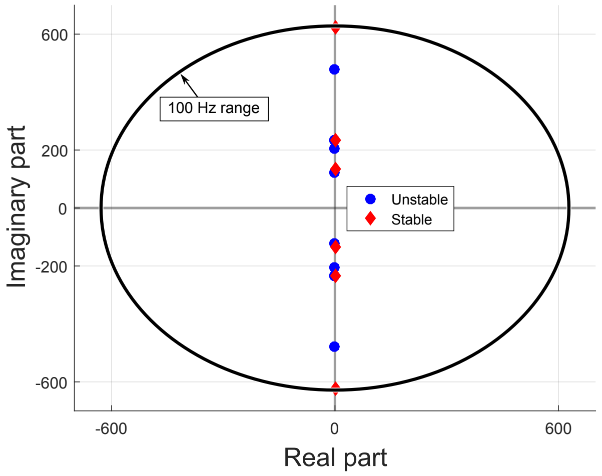


Figure 5.16: Poles of the linearized spatial parallel arm.

5.2.2 Results and discussion

Starting from the initial rigid solution, the optimization is completed after 12 iterations and lasts 89 minutes (5369 s). The resulting torques in the joints are shown in Fig. 5.17. One can observe a harmonic-like pre- and post-actuation in this case which is in accordance with the location of the poles close to the imaginary axis (see Fig. 5.16). As stated in the introduction, if the equilibrium points do not have hyperbolic behaviors, the duration of pre- and post actuation phases may not be finite. Indeed, with the present optimization formulation, a solution is found but the pre- and post-actuation phases seem to be rather long. Again, when the reference inputs $\mathbf{u}$ and $\mathbf{u}_{\text{rigid}}$ are used for in a direct dynamics simulation, the improvements done on the tracking performance is visible (Fig. 5.18(a)). When the rigid reference input $\mathbf{u}_{\text{rigid}}$ is used, the radius of the end-effector trajectory tends to drift away from the nominal radius (Fig. 5.18(b)). Again, this drift can be compensated for when flexibility is considered. The relative rms error e_{rms} drops from 5×10^{-2} to 4×10^{-4} when $\mathbf{u}$ is used: although the flexible members are actuated harmonically, the end-effector tracks the prescribed trajectory.

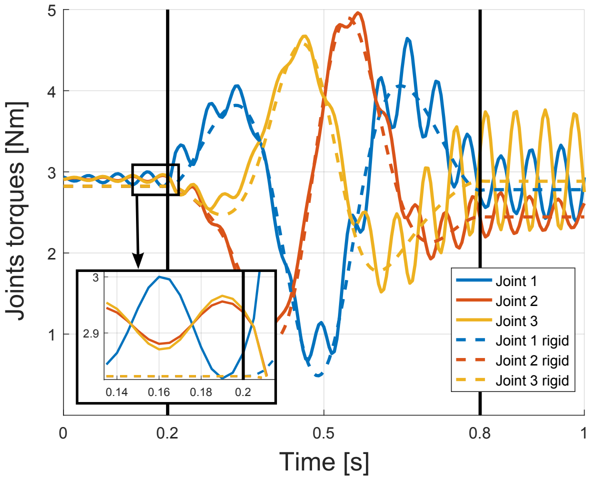
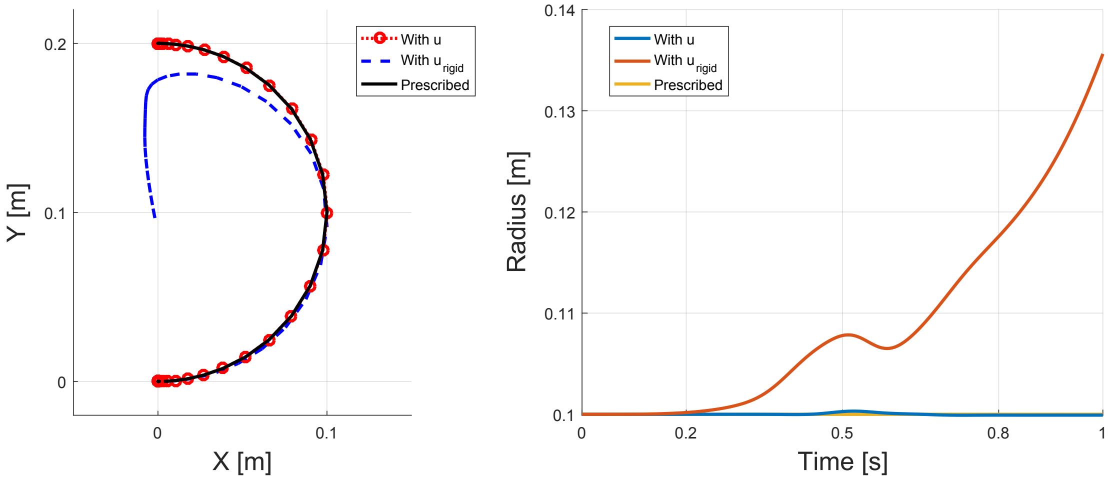


Figure 5.17: Joints torques of the flexible parallel 3D arm.



(a) End-effector trajectory of the flexible parallel 3D arm using $\mathbf{u}$ and $\mathbf{u}_{\text{rigid}}$.

(b) Radius of the resulting trajectory compared to the desired trajectory radius of $l_{low} = 0.1$m.

Figure 5.18: End-effector trajectory.

One important remark can be done for this parallel manipulator case. Since the poles of the system are close to the imaginary axis, considering some material damping in the system, as all real system has, the problem is much easier to solve. Indeed, if such damping is modeled as a Rayleigh damping in the beam using two parameters $\alpha = 1 \times 10^{-4}$ and $\beta = 1 \times 10^{-2}$, the poles of the system in the range of interest shift towards the stable left half plane as shown in Fig. 5.19. As a result, a time integration of the system from an initial condition converges to a bounded solution and no stable inversion technique is required.

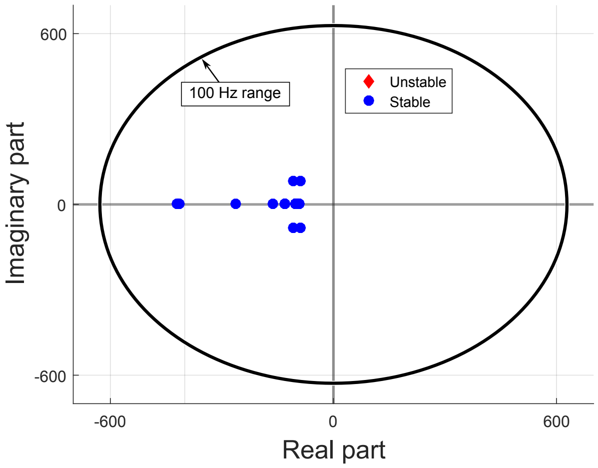


Figure 5.19: Poles of the linearized spatial parallel arm with Rayleigh damping.

5.2.3 Alternative architecture

In the first configuration of the parallel manipulator, when no material damping is considered, it is shown that the equilibrium points at the initial (and final) time are rather harmonic. This resulted in harmonic pre- and post-actuations periods that tend to be rather large. An alternative parallel architecture is now considered to illustrate the case of a parallel manipulator with hyperbolic equilibrium points.

Flexible upper links and rigid lower links

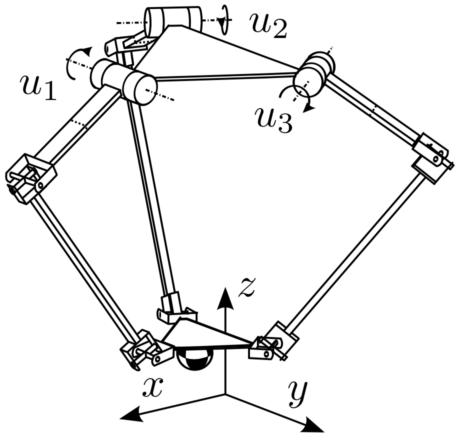


Figure 5.20: Parallel 3D arm system with flexible upper links and rigid lower links.

The upper links of the parallel robot, initially considered as rigid bodies, are now considered as flexible. The stiff tubular members are replaced by slender bodies with a rectangular cross section with a height b_1 much smaller than its width a_1. The lower links are considered as rigid bodies with full square cross sections of side a_2.

The height of the system and end-effector mass are kept the same. A schematic representation of this second parallel system is shown in Fig. 5.20. The material properties of the arms are the same as previously and the new geometric parameters are shown in Table 5.4.

Upper and lower plates	$l_{up} = 0.2$ m	$l_{low} = 0.2$ m	
Upper links	$l_1 = 0.6$ m $E = 70$ GPa	$a_1 = 0.1$ m $\nu = 0.3$	$b_1 = 0.0015$ m $\rho = 2700$ kg/m^3
Lower links	$l_2 = 0.4610$ m	$a_2 = 0.01$ m	$\rho = 2700$ kg/m^3

Table 5.4: Parameters of the parallel arm system with flexible upper links.

The prescribed trajectory is the same as defined in Eq. (5.2) with the new l_{low} parameter. With this new architecture where the flexible components are directly connected to the actuated joints, some poles of the system are now located on the real axis and the system has now some hyperbolic behavior, as shown in Fig. 5.21. The first pole is located at 9 Hz and is on the real axis. With the same tolerance and time step size as previously, the optimization process requires 9 iterations and lasts 77 minutes (4657 s). Note that the computation time is reduced to 11 minutes if a time step of 0.01 s is used. The flexible and rigid reference inputs are shown in Fig. 5.22. This time one can observe that the resulting input no longer oscillates harmonically as previously. Pre- and post-actuation of the system is also visible in this case. Again, the system starts on the first unstable eigenvector and ends on the first stable eigenvector as it is shown in Fig. 5.23.

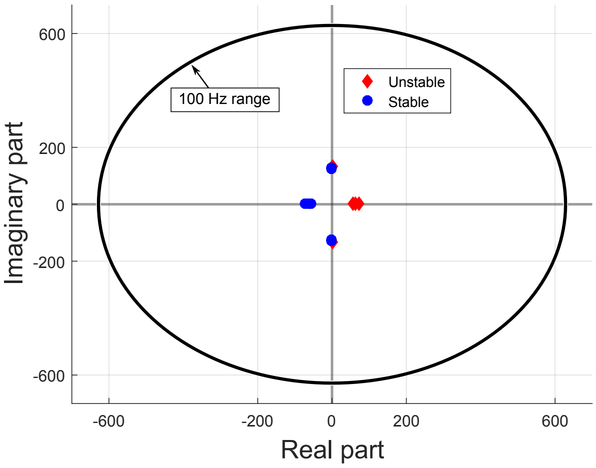


Figure 5.21: Poles of the linearized spatial parallel arm with flexible upper links.

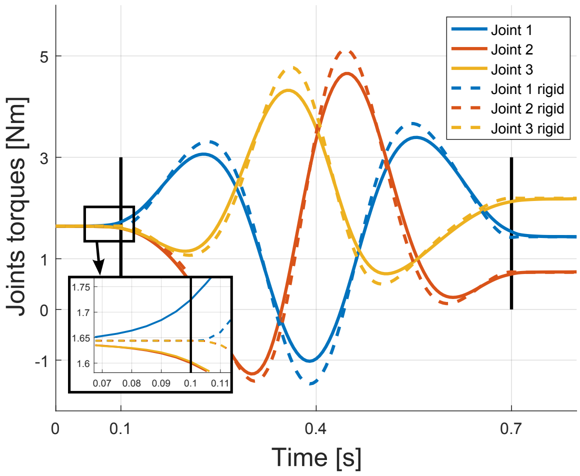


Figure 5.22: Joints torques of the parallel arm with flexible upper links.

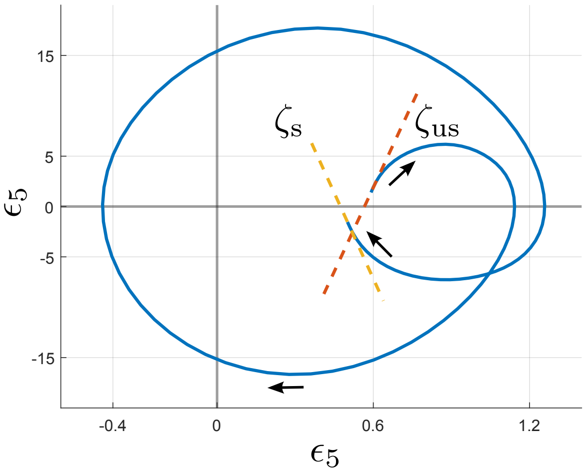


Figure 5.23: Phase plot of the fifth component of the deformation $\boldsymbol{\epsilon}$ (bending around local y axis) in the first beam element of the first upper link.

5.3 Summary

In this chapter, the inverse dynamics solver based on the optimization formulation is applied to spatial systems modeled using a FEM formulated on $SE(3)$. It is shown that the optimization method is able to deal with the more complex orientation representation of spatial problems. With the spatial serial examples, bounded feedforward inputs can indeed be computed although the flexible systems present non-minimum phase characteristics. Non-causality in the computed solutions is also found in the present example, although pre- and post-actuation may be quite small if the system is rather stiff. In the alternative architecture where the second link is more flexible, the poles of the system are closer to the imaginary axis and non-causality of the solution is clearly visible. Parallel manipulators can be treated in the same manner using the present formulation. In the parallel examples, without damping in the flexible members, the poles of the system are close to the imaginary axis and the hyperbolic equilibrium point hypothesis is arguable. Although a bounded solution is found, the resulting harmonic pre-actuation might not be realistic for real experiment since it tends to last for an infinitely long period of time. Considering the material damping of the system leads to a shift of the system to a minimum phase behavior. An alternative architecture where the flexible components are directly connected to the actuated joint is also investigated for the parallel case. This resulted in a non-minimum phase parallel system with hyperbolic equilibrium points at the initial and final time and finite pre- and post-actuation phases are obtained. These simulation examples all resulted in DAE of index 3.

Chapter 6

Experimental applications

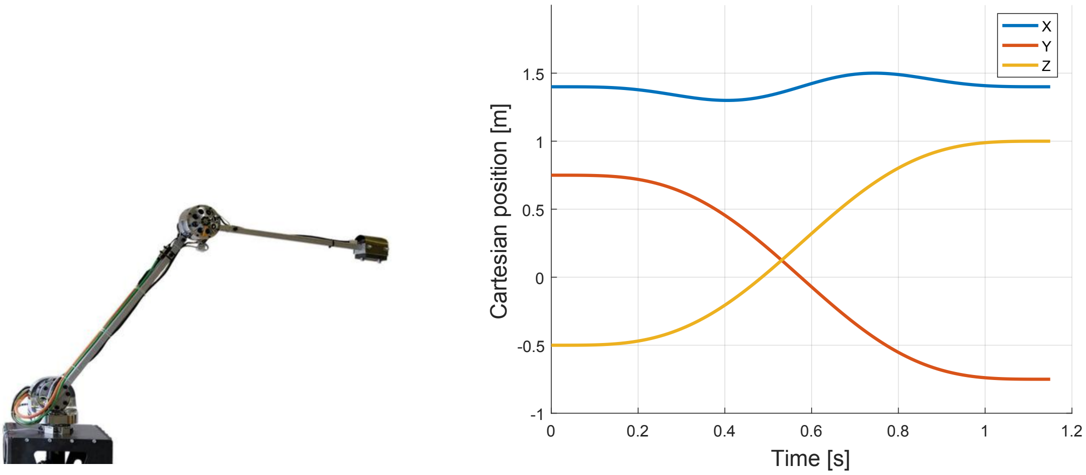
Usual robotic manipulators found on the market are designed in such a way that flexible dynamics behavior can be neglected. More recently, however, there has been an increasing interest for including flexible elements in robotic applications. The previous chapters demonstrated the potential of the inverse dynamics algorithm in numerical simulation. The present chapter studies the application of this method to experimental robotic systems and analyzes the resulting improvements in the dynamic performances.

6.1 ELLA robot

This first application considers a flexible link robot: the ELLA robot [111]. It is a 3 actuated dof robot built at the Institute of Robotics of the Johannes Kepler University Linz, Austria. It is made of two steel flexible links and 3 harmonic drives that allows 3D space motions (Fig. 6.1(a)). In this study, the desired Cartesian trajectory of the end-effector is shown in Fig. 6.1(b). The trajectory is roughly a straight line of about 2 m starting from the lower left side of the robot (positive y and negative z) and going to the upper right side of the robot (negative y and positive z). This trajectory has to be completed in 1.15 s. Commands are sent to each joint every 0.4 ms (frequency of 2.5 KHz) using an industrial computer.

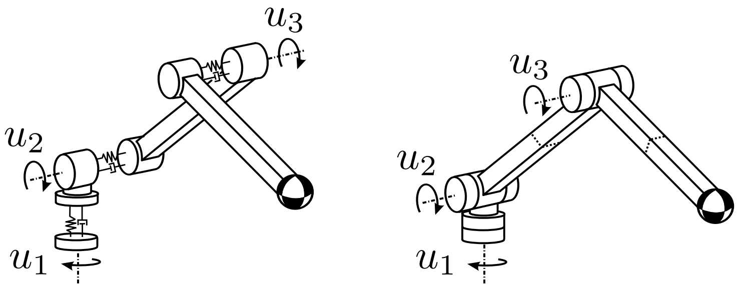
The first link is connected to the robot's base through the first and second joints which rotate around the vertical z axis and a horizontal axis respectively. The third joint, rotating around a horizontal axis parallel to the second joint axis, connects the first link to the second link of the robot. A 2,1 kg mass is fixed at its end-effector. The motor torques are transmitted through harmonic drive gears to the links. Generally speaking, these gears can have internal flexibility and friction.

Three different models are considered for the inverse dynamics analysis: a FEM model, a LEM model and a rigid model which are described hereafter.



(a) Picture of the ELLA robot. (b) Desired cartesian trajectory of the end-effector.

Figure 6.1: ELLA robot with 3 dofs and 2 flexible links.



(a) Lumped mass model. (b) Flexible beam model.

Figure 6.2: Two different models of the ELLA robot.

6.1.1 Joint friction

As stated in Section 1.2, friction inside a robotic joint can be represented using different models such as viscous, Coulomb, static and Stribeck models. For harmonic drives, it is reasonable to consider a combination of the first two models.

- Viscous friction force τ_v: this component assumes a friction force that is proportional to the joint relative velocity $\dot{\alpha}_J$. Mathematically, it is defined by a parameter d, equivalent to a viscous damper element and

$$\tau_\text{v} = \dot{\alpha}_J d \tag{6.1}$$

- Coulomb friction force τ_C: the friction force only depends on the sign of the relative velocity $\dot{\alpha}_J$ of the joint. It is assumed that once the joint is in relative motion, the force is constant. It is expressed as

$$\tau_C = f_C \text{sign}(\dot{\alpha}_J) \tag{6.2}$$

  where $f_C > 0$ is the maximum Coulomb friction force value.

In the present case, only the viscous friction in the joint is considered for two reasons. Firstly, the Coulomb friction is not a smooth function and thus result in a jump in the joint torques when their velocity crosses zero. It seems challenging to capture these jumps using the proposed inverse dynamics method and to implement such a nonsmooth control law on a real hardware system. Secondly, in simulation, the effect of Coulomb friction on the joint reference positions was investigated and appear to be negligible (order of 1×10^{-6} rad).

6.1.2 FEM model

In the FEM model, flexibility is only represented in the links and is modeled using non-linear beam elements as described in Section 3.5.2. The parameters describing the cross-section geometry and the material elastic properties are defined from the real manipulator data. The resulting model is depicted in Fig. 6.2(b) and the parameters are shown in Table 6.1. Each link is discretized into 2 elements, hence 3 nodes, and viscous friction is considered in the joints. As a consequence of the link flexibility, the resulting model has some internal dynamics. When analyzing the linear stability of the internal dynamics in the initial (and final) configuration, one finds that the system in non-minimum phase with the real part of its first unstable pole located at 24 Hz (see Fig. 6.3). The resulting DAE has index 3.

In order to compute the feedforward inputs, the inverse dynamics problem is solved using the optimization based approach, with the objective expressed using Eq. (4.31). The initial guess of the optimization is the inverse dynamics solution of an equivalent rigid link manipulator. The material damping is modeled as a Rayleigh damping in the beam through the α and β parameters. These coefficients have been tuned to fit the experimental behavior.

For the given desired trajectory, the inverse dynamics solution is computed in 280 seconds (10 ms time discretization using a x64 bits i7-4600u CPU with 16 Gb RAM memory). In order to control the joints at 2.5 KHz, the solution is interpolated on a refined time grid of 0.4 ms.

First link	$l = 0.9595$ m	$A = 3.01 \times 10^{-4}$ m	
	$I_{xx} = 5.9 \times 10^{-8} m^4$	$I_{yy} = 1.75 \times 10^{-8} m^4$	$I_{zz} = 2.1 \times 10^{-8} m^4$
Second link	$l = 0.9340$ m	$A = 2.09 \times 10^{-4}$ m	
	$I_{xx} = 2.85 \times 10^{-8} m^4$	$I_{yy} = 7.5 \times 10^{-9} m^4$	$I_{zz} = 1.01 \times 10^{-8} m^4$
Material	$E = 210$ GPa	$\nu = 0.3$	$\rho = 7800$ kg/m^3
	$\alpha = 1 \times 10^{-4}$	$\beta = 1.5 \times 10^{-2}$	
Visc. friction	$d_1 = 19.5$ Nms/rad	$d_2 = 22$ Nms/rad	$d_3 = 12.21$ Nms/rad

Table 6.1: Parameters of the ELLA FEM model.

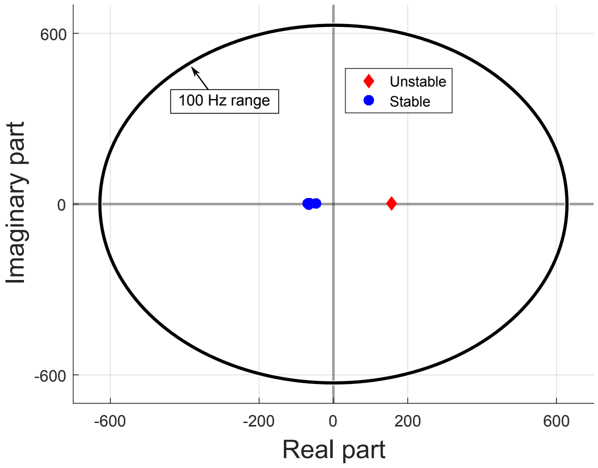


Figure 6.3: Poles of the linearized ELLA arm in its initial configuration with Rayleigh damping.

One can also be interested in the behavior of the internal dynamics in this case. Similarly to the serial arm examples of Section 5.1, the fifth component of the deformation $\boldsymbol{\epsilon}$ of the first beam element (bending around local y axis) is considered in the $(\epsilon_5, \dot{\epsilon}_5)$ plane. As previously, the dynamics is driven from the neighborhood of the first unstable eigenvector to the neighborhood of the first stable eigenvector.

6.1.3 LEM model

An alternative model used to generate the feedforward command is a LEM model (see Fig. 6.2(a)). This model is explained in detail in reference [111] and is only summarized here. In this model, links are considered as rigid bodies and the flexible behavior of the system is taken into account through the 3 virtual spring elements located at the joints of the robot. The parameters of these elements are identified from the experimental measurements. For each joint, there is a motor related angle q_M and a link related angle q_L, the distinction between these two angles being

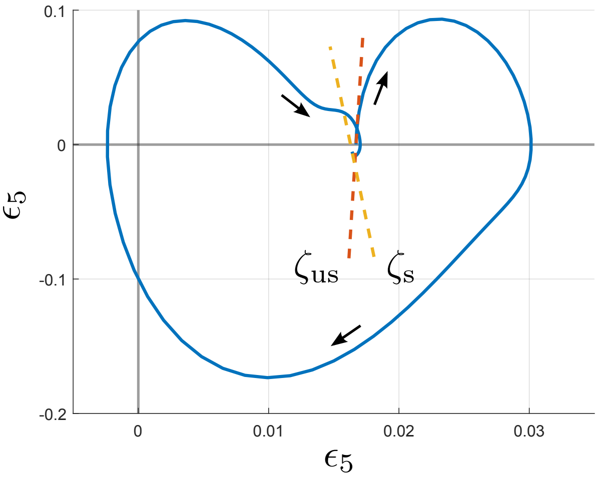


Figure 6.4: Phase plot of the fifth component of the deformation $\boldsymbol{\epsilon}$ (bending around local y axis) in the first beam element of the ELLA robot.

a consequence of the flexible joint hypothesis. The parameters of the model are shown in Table 6.2.

First link	$l = 0.9595$ m	$m = 2.25$ kg	
	$J_{11} = 5.18 \times 10^{-4}$ kgm^2	$J_{22} = J_{33} = 0.1625$ kgm^2	
Second link	$l = 0.9340$ m	$m = 1.42$ kg	
	$J_{11} = 2.18 \times 10^{-4}$ kgm^2	$J_{22} = J_{33} = 0.08374$ kgm^2	
Stiffness	$k_1 = 12$ kNm/rad	$k_2 = 13$ kNm/rad	$k_3 = 7.5$ kNm/rad
Visc. friction	$d_1 = 19.5$ Nms/rad	$d_2 = 22$ Nms/rad	$d_3 = 12.21$ Nms/rad

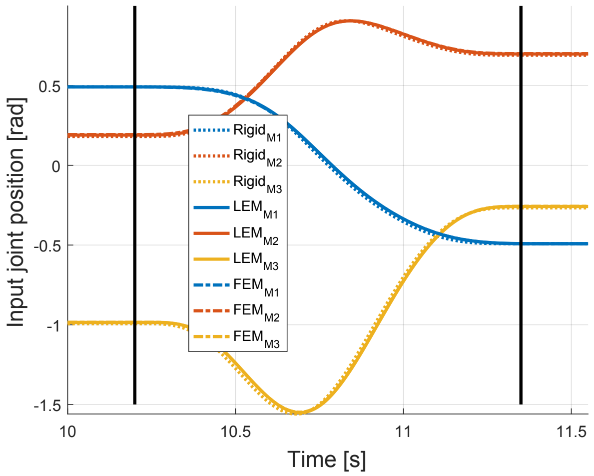
Table 6.2: Parameters of the ELLA LEM model.

Since the links are considered as rigid, the desired trajectory at the end-effector can easily be written in terms of the link angle $q_{L,n}$ of each joint n. If these link angles $q_{L,n}$ are now considered as the new outputs, the system is flat and the feedforward command for the 3 motor angles $q_{M,n}$ can be computed algebraically at each time step based on the flatness property of the system. This model, implemented in Simulink, is used to compute an inverse dynamics solution of the system in real time for a sampling rate of 2.5 kHz [111].

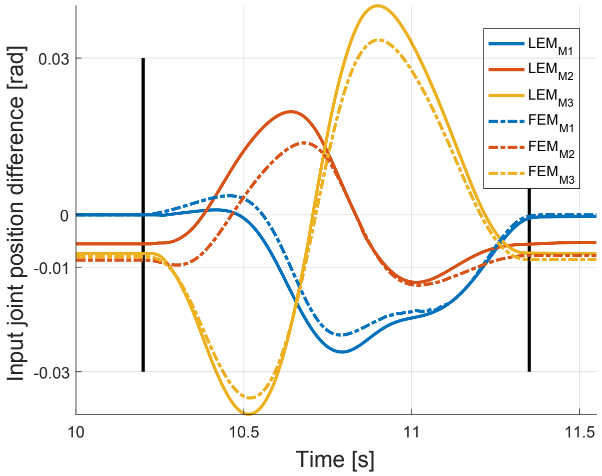
6.1.4 Rigid model

The rigid model considers rigid links and joints without any elasticity in the transmission chain: the joint angles are equal to the link angles and links are again modeled as rigid bodies. Note that viscous friction inside the joints is still considered here. The solution of the inverse dynamics of this model can be computed in real time using the flatness based approach.

6.1.5 Comparison of the reference trajectories



(a) Joint reference trajectories $\mathbf{q}_{\text{in}}$.



(b) Differences between LEM and FEM reference trajectories compared to rigid reference trajectory.

Figure 6.5: Comparison between reference trajectories $\mathbf{q}_{\text{in}}$ resulting from the flexible (LEM and FEM) and rigid models.

From the inverse dynamics solution, one can obtain the reference trajectories $\mathbf{q}_{\text{M,FF}}$ from the models described above: (1) the FEM model solved using the optimization based approach, leading to the *FEM reference trajectory*, (2) the LEM model solved using the flatness based approach, leading to the *LEM reference trajectory*, and (3) the rigid model solved using the flatness based approach, leading to

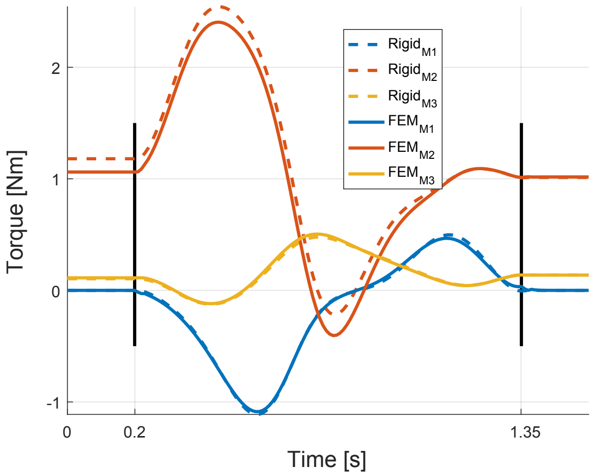
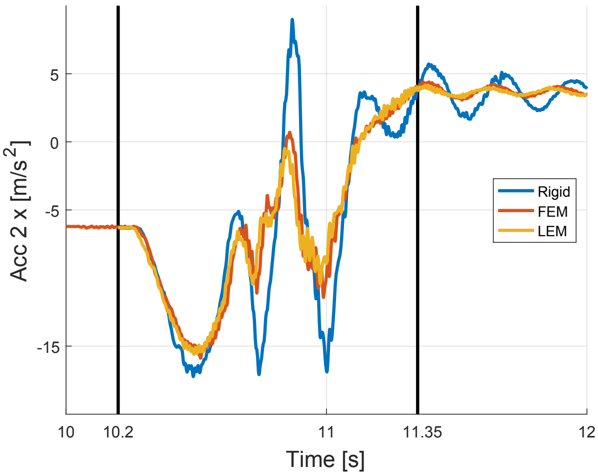


Figure 6.6: Computed reference torques $\mathbf{u}_{\mathrm{FF}}$ from the FEM and rigid models.

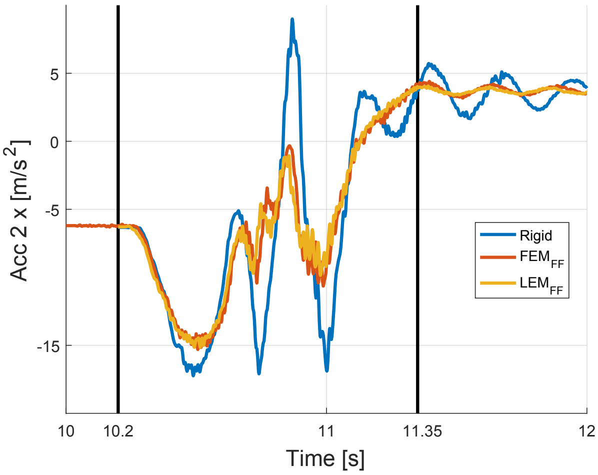
the *rigid reference trajectory.* In Fig. 6.5(a), the three references trajectories $\mathbf{q}_{\mathrm{M,FF}}$ are compared. The vertical black lines indicate when the end-effector trajectory starts and ends. One can observe that these reference trajectories have some slight differences (see Fig. 6.5(b)) of a few hundredth of a radiant compared to the rigid reference. Flexibility in the system also leads to an initial correction in the joint position to compensate for the initial static deflection due to gravity. The inverse dynamics solution also provides the reference torques $\mathbf{u}_{\mathrm{FF}}$. Here, only the reference torques computed from the FEM and rigid model are shown in Fig. 6.6. Again, only slight adaptations from the rigid solution are required. Similarly to the first serial arm example, pre- and post-actuation of the system is hardly visible. Again, it is observed that the manipulator in this case is rather stiff so that the amplitude of the internal dynamics is small.

6.1.6 Experimental results

Experimentally, two strategies are tested. The first one only feeds the reference trajectories $\mathbf{q}_{\mathrm{M,FF}}$ to the system. The second strategy feeds both reference trajectories $\mathbf{q}_{\mathrm{M,FF}}$ and reference torques $\mathbf{u}_{\mathrm{FF}}$ to the system. In both cases, a cascaded linear PD feedback loop is designed for each joint independently based on motor encoder signals. In order to evaluate the efficiency of the feedforward inputs, the vibrations of the robot are measured using two IMUs fixed on the robot arm: one at the end of the first link and one at the end-effector.



(a) Measured acceleration resulting from the reference trajectory $\mathbf{q}_{\text{in}}$ only.

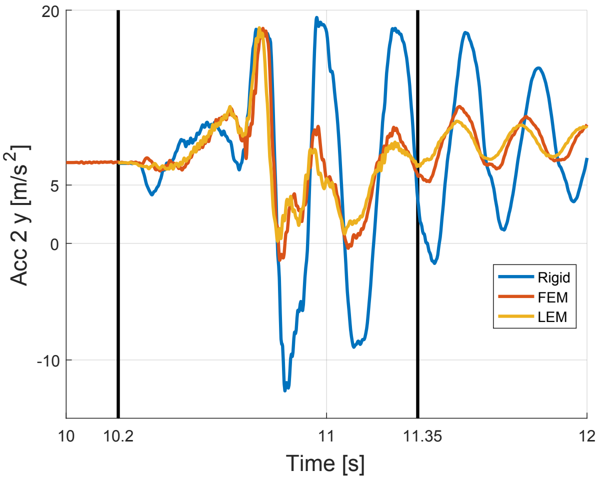


(b) Measured acceleration resulting from both the reference trajectory $\mathbf{q}_{\text{in}}$ and reference torques $\mathbf{u}_{\text{FF}}$.

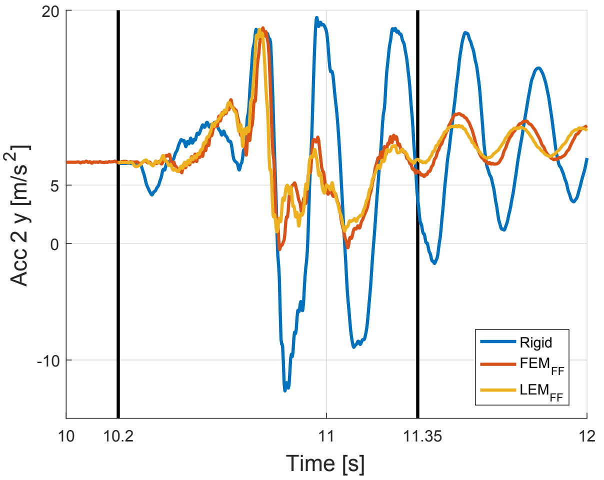
Figure 6.7: Measured acceleration at the end-effector of the ELLA robot in the local x direction of the IMU.

Feedforward of the reference trajectories $\mathbf{q}_{\text{M,FF}}$ alone

The measured accelerations at the end-effector are shown in Fig. 6.7(a) and Fig. 6.8(a) in the x (along the beam axis) and y (transverse to the beam) directions of the links respectively. The first rigid curve is the resulting acceleration when the robot is controlled with the rigid reference trajectory. As can be expected, the amplitude of the acceleration during the motion is largely reduced when a flexible behavior is considered. Once the robot reaches its final position, some residual vibrations remain in the tip. Again, these residual vibrations are considerably reduced when the FEM or LEM reference trajectories are used (around 50% of the amplitude in the y direction). In Fig. 6.8(a), the accelerations reach an upper bound simply because



(a) Measured acceleration resulting from the reference trajectory $\mathbf{q}_{\text{in}}$ only.

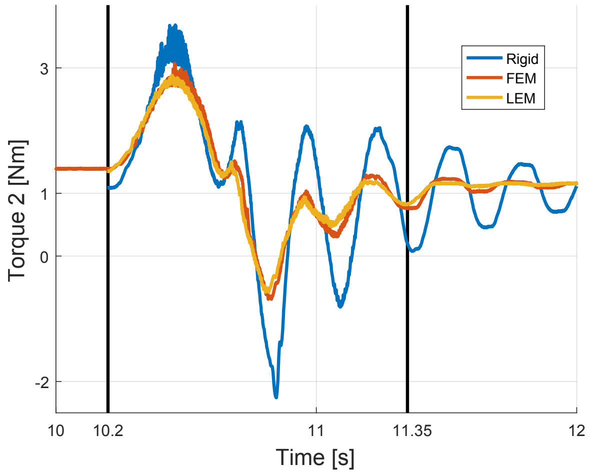


(b) Measured acceleration resulting from both the reference trajectory $\mathbf{q}_{\text{in}}$ and reference torques $\mathbf{u}_{\text{FF}}$.

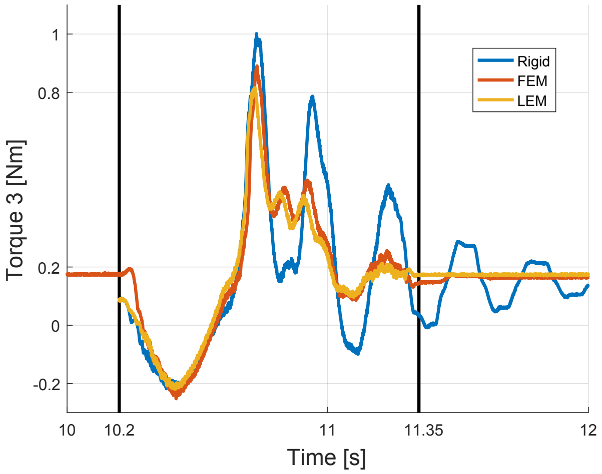
Figure 6.8: Measured acceleration at the end-effector of the ELLA robot in the local y direction of the IMU.

the accelerometer saturates around 20 m/s^2. From the observation of these figures, the FEM and LEM models lead to very similar behavior and tracking accuracy of the manipulator.

The resulting torques applied on each joint of the system can also be measured. These are shown for the last two joints in Fig. 6.9(a) and Fig. 6.9(b). In both cases, the oscillating behavior resulting from the feedback control is visible during the motion. However, when the flexible reference trajectories are used, the torques required to drive the system are reduced. For the third joint, the torque almost remains constant after the end of the trajectory; almost no correction is required from the feedback.



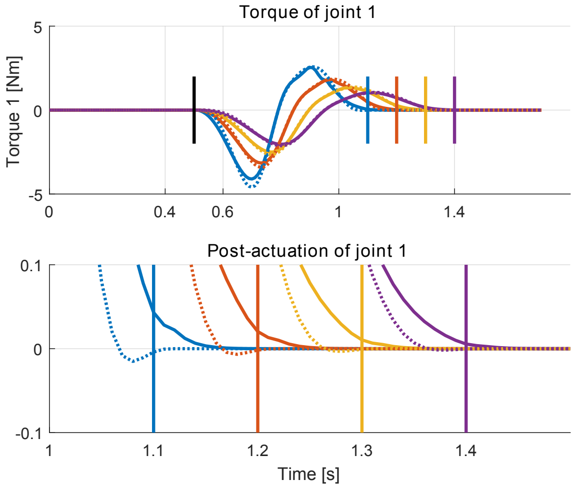
(a) Measured torques in the second joint.



(b) Measured torques in the third joint.

Figure 6.9: Measured torques on the second and third joints of the ELLA robot resulting from a control with the reference trajectories $\mathbf{q}_{\text{in}}$ only.

Note For all models, geometric parameters, such as cross sections and length of the links, were measured on the real manipulator. The cross section and material parameters, needed for the FEM model, were based on real data and were not calibrated to fit the experimental response. In contrast, the stiffness and damping parameters of the flexible joints of the LEM model had to be identified additionally. This identification process may explain the slight improvement on the end-effector trajectory resulting from the LEM reference trajectory compared to the FEM reference trajectory.


Torque of joint 1

Post-actuation of joint 1

Figure 6.10: Influence of the trajectory tracking time on the post-actuation of the first joint of the ELLA robot.

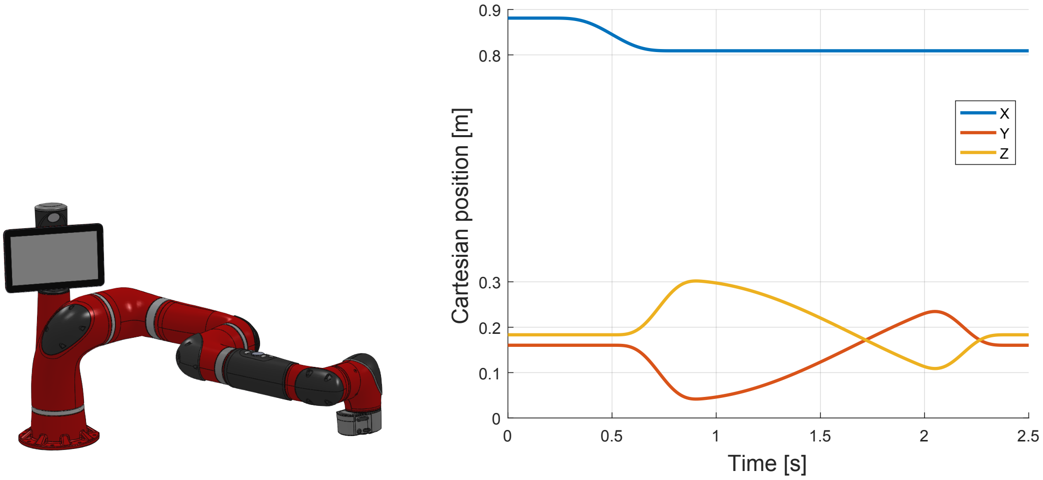
Feedforward of the reference trajectories $\mathbf{q}_{\mathrm{M,FF}}$ and torques $\mathbf{u}_{\mathrm{FF}}$

With this second strategy, additional reference torques $\mathbf{u}_{\mathrm{FF}}$ from the FEM or rigid model are provided to the controller. The resulting accelerations at the end-effector in the x and y directions are shown in Fig. 6.7(b) and Fig. 6.8(b) respectively. One can see that no additional improvements on the vibration are done in this case. It is assumed that the PD feedback gains may need to be adapted when this additional feedforward contribution is considered. Also, Coulomb friction effects and other model discrepancies inside the joints may not be negligible anymore if further improvement is wanted. In this work, the model of the robot is not developed further.

6.1.7 Simulation of more aggressive trajectories

As noted in Section 1.1.2, references [40, 100, 5] state that the inversion of non-minimum phase system leads to non-causal solutions: the system may need to be actuated before (or after) the prescribed trajectory actually starts (or ends). In the current application, although the computed inputs lead to a reduction of the flexible vibration in the system, no pre- or post-actuation of the system is visible when looking at the reference torques. Similarly to the simulated examples of Section 5.1, in this case, the system is rather stiff and do not require such non-causal actuations with the given trajectories. Using the FEM model, more aggressive trajectories are carried out in simulation. The aim is to see if pre- or post-actuation of the system would be needed in such cases. The starting and ending point of the end-effector trajectory are the same as previously. However, the trajectory should now be tracked in 0.9, 0.8, 0.7 or 0.6 s as shown in Fig. 6.10. In this figure, the continuous lines represent the flexible reference torques of the first joint whereas the doted lines are the rigid reference torques. The vertical lines represent the start (at 0.5 s) and the end of the tracked trajectory for each tracking time. One can note that as the trajectory becomes more aggressive i.e. the tracking time decreases, the post actuation on the joint becomes more noticeable. On the other hand, these more aggressive trajectories do not lead to noticeable pre-actuation of the system. Similar observation can be done for the two other joints. To conclude, since pre-actuation is related to the unstable characteristics of the system, it is assumed that the present system has its unstable poles far enough not to be excited. These trajectories were not experimentally tested on the real ELLA robot because they would require more important actuating torques, that can not be achieved by the motors of the robot.

6.2 Sawyer robot

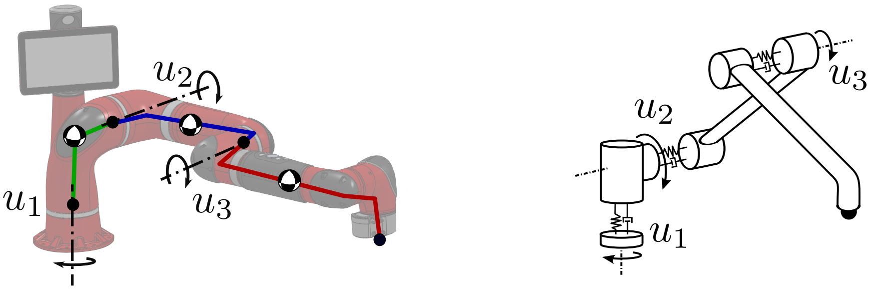


(a) Sawyer robot (b) Desired cartesian trajectory of the end-effector.

Figure 6.11: Sawyer robot with flexible joints and 7 dof.

The Sawyer robot is a 7 dof serial collaborative robot (see Fig. 6.11(a)) built with serial elastic actuators [93] that include flexure springs. As a result, flexibility of the joints strongly influence the dynamic response of the arm. Its links are made out of aluminum and are fairly rigid. The Intera interface is used to control the robot. This interface, based on the open source Robot Operating System (ROS), allows to use low level commands of the joints position, velocity and torques at a maximum rate of 800 Hz. In the present test case, the prescribed end-effector trajectory that is considered is shown in Fig. 6.11(b).

6.2.1 Sawyer model



(a) Simplified 3 actuated dof Sawyer robot.

(b) Schematic model with flexible joints.

Figure 6.12: Simplified 3 actuated dof Sawyer robot with flexible joints.

For this case study, the Sawyer robot is modeled as a 3 actuated dof serial robot with rigid links and flexible joints (as represented in Fig. 6.12(a)): joints 3, 5, 6 and 7 remain fixed and only joints 1, 2 and 4 are actuated. For the sake of clarity, the latter is renumbered as joint 3. This leads to the actuating torques u_1, u_2 and u_3 as shown in Fig. 6.12(b)). It is important to note that as a consequence of the flexible joint model, the resulting DAE describing the motion of the system no longer has index 3, i.e. the matrix defined in Eq. (4.17) becomes singular in this case.

6.2.2 Model identification

First joint	$k_1 = 750$ Nm/rad	$d_1 = 10$ Nms/rad
Second joint	$k_2 = 1000$ Nm/rad	$d_2 = 5$ Nms/rad
Third joint	$k_3 = 450$ Nm/rad	$d_3 = 1$ Nms/rad

Table 6.3: Joint parameters used for the Sawyer arm.

The flexure springs in the serial elastic actuators, which give an intrinsic flexibility to the joints, are modeled using torsional springs with a linear behavior connecting two consecutive rigid bodies. Since the stiffnesses k_i of spring i are not provided by the robot manufacturer, they are identified using the torque sensing capabilities of the robot. The robot arm is commanded to remain in its Gentry pose

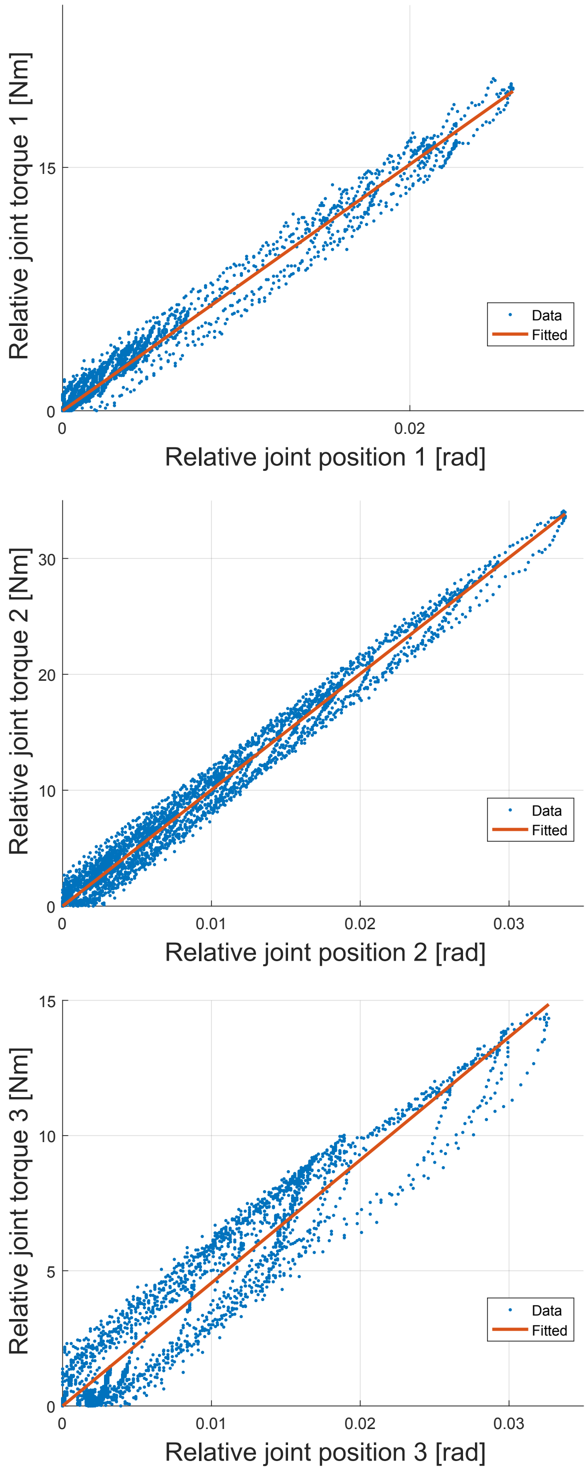


Figure 6.13: Identification of the joints stiffnesses of the Sawyer robot.

(see Fig. 6.11(a)) during the whole identification process. Each joint is twisted manually to generate torsion in the flexure springs. The torque increase on each joint is then plotted with respect to the displacement of each joint as shown in Figs. 6.13.

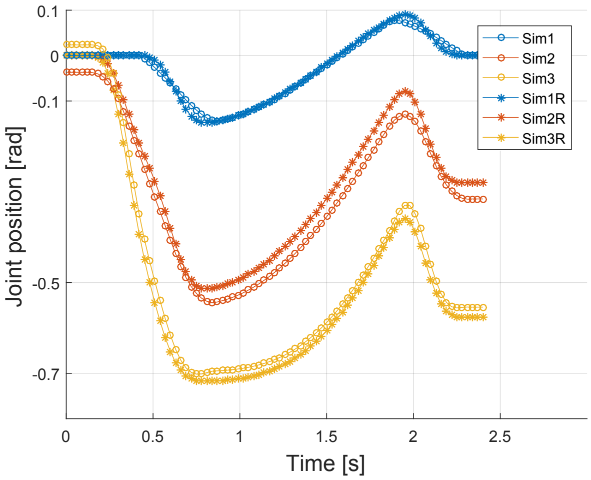
A linear fitting of the data points in the least square sense is then performed to obtain the linear stiffness coefficient. Looking into the torque-displacement curve of the third joint, one can notice that some non-linear effects, such as hysteresis or friction, are also present in the springs. However, those are not considered in the present identification process. The flexible joint properties are summarized in Table 6.3. Unlike the spring stiffnesses, viscous friction parameters d_i of each joint i and inertia properties of the rigid bodies are defined thanks to the models and CAD files provided by the robot manufacturer.

6.2.3 Results and discussion

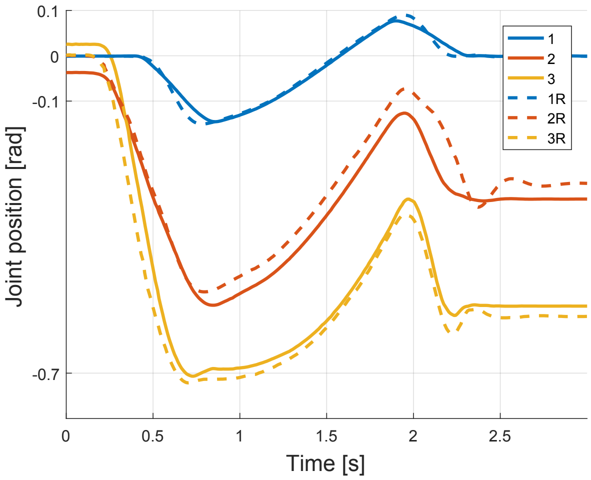
As noted in the introduction (Section 1.1.2), the inverse dynamics of flexible joint systems do not require the use of stable inversion techniques. However, here the aim is to show that the proposed approach is general and able to deal with various types of flexible systems, a flexible joint system in this case. The inverse dynamics problem is thus also solved using the optimization based approach where the objective function is expressed using the relative coordinate of each joint as in Eq. (4.29). The initial guess to start the optimization is the solution of the inverse dynamics of an equivalent rigid arm i.e., an arm where the flexible joints are locked. The computation of the inverse dynamics solution takes 4 iterations and lasts around 505 s (10 ms time discretization using a x64 bits i7-4600u CPU with 16 Gb RAM memory laptop). The solution is interpolated on a finer time grid of 2 ms in order to control the arm at a reliable rate of 500 Hz.

The joint reference trajectories $\mathbf{q}_{\mathrm{M,FF}}$ computed with the flexible model are shown in Fig. 6.14(a) with circular markers. In what follows, it will be referred to as the flexible reference trajectory. It is compared to the reference trajectory computed with the rigid model (star markers), referred to as the rigid reference trajectory. One visible difference is the offset at the beginning of the trajectory which is the static gravity compensation in the second and third flexible joints. Around 0.5 s, one can observe that the first joint starts its motion a bit earlier when flexibility is considered. A similar observation is seen at the end of the trajectory.

These reference trajectories are sent to the robot controller. The actual joint positions are measured by the internal encoders of the robot and shown in Fig. 6.14(b). While almost no vibration is observed with the flexible reference trajectory (continuous line), the use of the rigid reference trajectory leads to residual vibrations in the second and third joints at the end of the trajectory (at around 2 s). These vibrations are the result of the feedback action trying to correct the actual joint



(a) Joint reference trajectories.

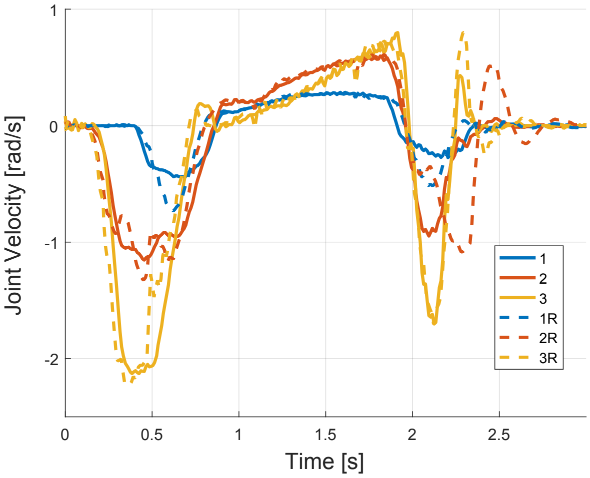


(b) Joint positions measured on the real system.

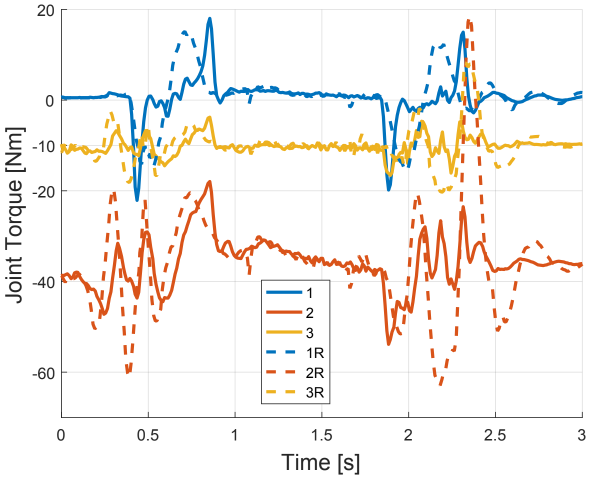
Figure 6.14: Computed and measured joint inputs.

positions. Looking at the measured joint velocities of each joints (see Fig. 6.15(a)), such vibrations are even more visible around the end of the tracking. It is also interesting to notice in Fig. 6.15(b) that the efforts required to drive the arm are largely reduced, in particular for the second joint. As a result, less energy is involved with the flexible reference trajectories.

For this application, the resulting end-effector trajectory is measured using a *Codamotion* optical tracking system (Charnwood Dynamics; Rothley, 62 UK) capturing infrared (IR) signals emitted from active markers. Four CX1 Coda sensor units (Fig. 6.16(a)) are placed at strategic locations around the robot as shown in Fig. 6.16(c) (3 sensor unit are visible). This allows a continuous tracking of each marker at an acquisition frequency of 200 Hz. The Codamotion system can track the position of each marker down to a 0.1 mm accuracy. The active markers are



(a) Measured joint velocities.

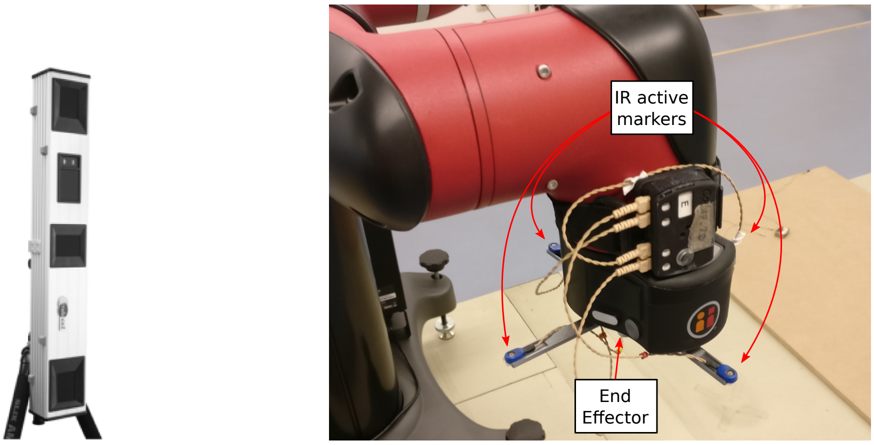


(b) Measured joint torques.

Figure 6.15: Measured joint data from the Sawyer robot.

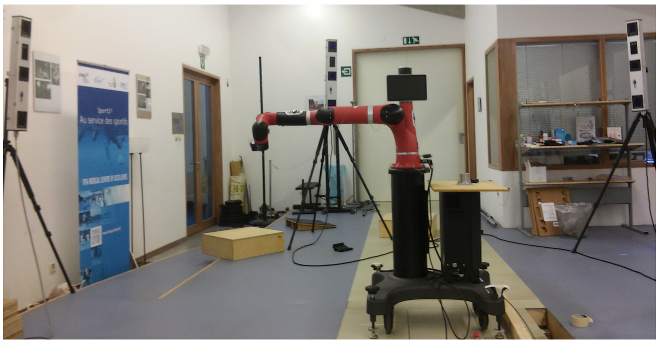
positioned on the robot's base and end-tool (4 on the basis and 4 on the end-tool) as shown in Fig. 6.16(b). The end-effector position can then be computed knowing the geometry of the end-tool.

The measured end-effector position is then plotted in Fig. 6.17. When the rigid reference trajectory is used, the effective trajectory of the arm is shifted along the z axis because of gravity which is not compensated in this case. A rather large deviation from the desired trajectory is observable around $y \in [0.15; 0.25]$. This corresponds to the end of the trajectory (around 2 s) where the oscillations were more important. The RMS relative error done on the tracked trajectory is $3,8 \times 10^{-2}$. One can observe that with the flexible reference trajectory, the effective trajectory has smaller deviations. The RMS relative error on the tracked trajectory is $2,4 \times 10^{-2}$ in this case. The error could be further reduced by identifying more precisely the



(a) CX1 Coda sensor unit.

(b) Active IR markers on Sawyer's end-effector.



(c) Picture of the setup in the Laboratory of Human Motion Analysis (LAMH) of the University of Liège.

Figure 6.16: 3D position tracking setup for the Sawyer's end-effector.

effective inertias and flexible joint parameters of the arm.

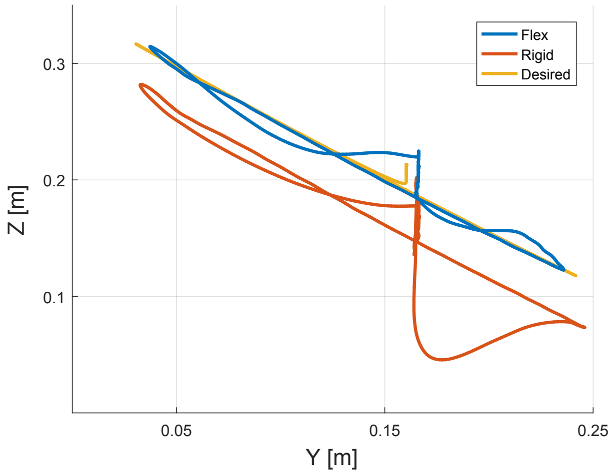


Figure 6.17: Trajectory comparison of the end-effector of the Sawyer robot in the $y-z$ plane.

6.3 Summary

In this chapter, the experimental inverse dynamics problem of a flexible link robot and a flexible joint robot is studied. Regardless of the flexible characteristics, a stable inversion procedure based on the minimization of the internal dynamics (or elastic deflection) allows to find a sensible solution to the inverse dynamics problem of the present robots. When applied experimentally to the manipulators, the resulting reference trajectories reduce the end-effector oscillations.

Thanks to a LEM model of the ELLA robot, it was possible to make a preliminary comparison between the FEM and LEM reference trajectories. It appeared that both led to similar reductions in the oscillations of the arm. The LEM model leads to an algebraic problem which allows the inverse dynamics to be solved in real time. Therefore, it is quite useful for control purposes. However, the modeling procedure involves the identification of equivalent spring and damper parameters to model link flexibility. The generalization of this procedure to more complex 3D problems may become challenging. On the other hand, the FEM model leads to a higher dimensional problem which can only be solved off-line. However the model was constructed only based on material and geometrical properties of the manipulator without any calibration procedure. In the experiments, it is observed that the internal dynamics is not significantly excited, which explains the good results obtained using the LEM model. It would be interesting to investigate experimentally more aggressive trajectories where the internal dynamics would play a more important role and compare the results from both FEM and LEM models.

With the Sawyer robot, it was shown that flexible joint robots could be dealt with the optimization based solver in the same manner as the flexible link case. Once the flexible joint parameters were identified, the flexible reference trajectory allowed to reduce oscillation in the end-effector of the robot. It was verified using a position tracking system that the error on the tracked trajectory could indeed be reduced when flexible considerations were taken.

Chapter 7

Conclusions and perspectives

7.1 Summary

Lightweight and flexible robotic manipulators have a complex dynamic behavior as the vibrations may disturb the end-effector accuracy. Researches dealing with the modeling and control of such robotic arms are reported in the literature, especially for planar and open chain systems. The present thesis proposes a general approach for the modeling and inverse dynamics analysis of spatial manipulators. It is shown how the resulting numerical tool can be used to design feedforward controls for flexible 3D manipulators.

The modeling method is based on a general FEM approach for flexible MBS. The method relies on a formulation of the kinematics and dynamics of the system in the $SE(3)$ matrix Lie group where the motion of the system is represented by 4×4 transformation matrices. Thanks to a local frame representation of the velocities and strains, this formalism leads to reductions in the non-linearity of the equation of motion of the MBS that undergoes large displacements. Also, it has no restriction on the serial or parallel kinematic topology of the system.

The inverse dynamics problem is also formulated on the $SE(3)$ group by imposing servo-constraints. As one is interested to solve the inverse dynamics of systems which can present a non-minimum phase internal dynamics, the present method relies on a stable inversion technique. The inverse dynamics problem is stated as a constrained optimization problem where an objective function is minimized. The equations of motion and the prescribed trajectory are seen as constraints of the optimization problem. The resulting inverse dynamics problem is defined as a set of DAE on a Lie group. In order to use classical optimization tools which are defined for linear design variables, the non-linear design variables of the present problem are reformulated on the linear Lie algebra of the $SE(3)$ group. A direct transcription method is used to solve the resulting large but sparse NLP. The complete solver is integrated in the GECOS development code in Matlab® and is validated numerically on some spatial examples.

The first numerical examples illustrate the ability of the present method to deal with serial and parallel kinematic flexible systems in a common fashion. The generality of the approach allows one to solve the inverse dynamics of flexible MBS regardless of the existence or stability of their internal dynamics. The latter is strongly affected by geometric and material characteristics of the considered MBS. When the internal dynamics is unstable, the input drives the dynamics of the system from the unstable manifold to the stable manifold. Depending on the prescribed trajectory and the stiffness of the system, the presence of pre- and post-actuation can be more or less significant.

The method is then confronted experimentally on two real spatial manipulators. The first manipulator is the ELLA robot, a three actuated dofs serial manipulator with two elastic links. Two flexible models of the system are built based on the FEM and LEM respectively. Although, the former captures the non-minimum phase behavior of the real system, both inverse dynamics solutions lead to a similar improvement on the end-effector vibrations: up to 50% of reduction in the amplitude of the vibration in the y direction. The actuating effort in the manipulators was also reduced thanks to the optimized feedforward action. The second manipulator is a flexible joint industrial robot, the Sawyer robot. A flexible joint model is built and torsion stiffness parameters are identified for its joints. Although the system does not have an internal dynamics, the optimization problem formulation does not need to be adapted and a solution to the inverse problem is found. Experimentally, the end-effector position is monitored using an optical tracking system and it is shown that the relative error on the tracking trajectory decreases by almost 40%.

As a concluding remark, the present methodology can be used for the inverse dynamics analysis of flexible 3D manipulators. It allows the generation of improved feedforward inputs that reduce the vibrations inside the system. It is a generic approach able to find a bounded solution for a large class of systems, including non-minimum phase systems.

The main contributions of the thesis can be summarized as:

- To show the application of a geometric FEM modeling method to robotic systems.
- To demonstrate how such geometric approach can be extended to the formu-

lation of the inverse dynamics problem as a constrained optimization problem.

- To illustrate the impact of the internal dynamics on the inverse dynamics solution of simple linear and more complex spatial systems.

- To demonstrate the experimental control of flexible spatial manipulators using the present method.

7.2 Perspectives

Several perspectives and open questions can be discussed at the end of the present work.

- In the current work, a direct transcription method is used to solve the optimization problem. This results in a simple and robust formulation of the optimization problem. However, this also leads to an increasing dimension of the problem as the time grid is refined. It was shown in [22] that a shooting method could allow to refine the time grid of the simulation without increasing too much the optimization dimension. Following the ideas presented in [4, 13, 19], the adaptation of such shooting methods to solve the inverse dynamics of 3D spatial systems could be investigated.

- Up to now, the solver was implemented in Matlab®. This environment simplifies the implementation and validation process of the method for spatial MBS. As a result, the computation time required to solve the inverse dynamics prevented its integration in a real time controller. The implementation of such method in a compiled language could improve the numerical performances for that purpose.

- In flexible MBS analysis, when the dimension of a model is large, model reduction techniques are often used to reduce computational efforts. The integration superelement [107] or global modal parameterization (GMP) [25] techniques in the present geometric model could be investigated to further accelerate the numerical solver.

- Additionally, in order to approach real time constraints, model predictive (MPC) scheme could be investigated and the inverse dynamics could be solved with this optimization method on a reduced moving time horizon.

- The generality of the FEM approach allows to integrate other elements that could be useful for the analysis of flexible MBS and their interaction with the environment. For example, contact elements [20] could be integrated in the model to investigate elastic deflection resulting from the interaction with a defined environment. More detailed friction models could also be included. Feedback control inside the model has not yet been considered inside the optimization process. It would also be interesting to investigate the effect of feedback control in the optimization process.

- The performances of the controller are affected by the accuracy of the robot model. In order to have a model that better represents the real dynamics of the manipulator, its parameters can be calibrated and identified experimentally [61]. In future work, calibration and experimental identification methods for the modeling parameters could be investigated. The resulting performances of the solution could be compared for different modeling approaches.

- The interest for soft robotics is growing in the robotics community. The experimental application of the present method to more flexible or soft robots could be investigated. The stability of the internal dynamics may have an impact on the control of such robots, which could be further studied.

- In order to design efficient mechanical systems, structural optimization have been proposed for example in [114, 115]. These techniques aim at refining some design parameters in order to improve the performance and/or reduce the weight of the machine. It turns out that the optimal design is quite sensitive to the loading conditions and reference trajectory. Conversely, the inverse dynamics problem is also sensitive to the mechanical parameters so that the structural and control design problem are inherently coupled. In this context, it would be interesting to formulate a coupled optimization problem, integrating both the structural and control parameters, and to develop appropriate solution methods.

- On a more fundamental point of view, one could also investigate more deeply the impact of spatial discretization techniques for systems with internal dynamics. Indeed, in the analysis of the axial vibration of a bar, the minimum phase or non-minimum phase nature of the system is altered when standard FEM discretization is used. For applications dominated by axial deflections, specific discretization schemes [64] for wave propagation could be more appropriate.

Appendix A

Additional mathematical developments

A.1 Cross derivative and Lie bracket

The cross derivative of an element $\mathbf{H}$ of a Lie group is commutative and leads to

$$\begin{aligned}
\delta(\mathbf{H}\tilde{\mathbf{v}}) &= (\mathbf{H}\widetilde{\delta\mathbf{h}})^{\cdot} && \text{(A.1)}\\
\mathbf{H}\widetilde{\delta\mathbf{h}}\tilde{\mathbf{v}} + \mathbf{H}\delta(\tilde{\mathbf{v}}) &= \mathbf{H}\tilde{\mathbf{v}}\widetilde{\delta\mathbf{h}} + \mathbf{H}(\widetilde{\delta\mathbf{h}})^{\cdot} && \text{(A.2)}\\
\mathbf{H}\left(\widetilde{\delta\mathbf{h}}\tilde{\mathbf{v}} + \delta(\tilde{\mathbf{v}})\right) &= \mathbf{H}\left(\tilde{\mathbf{v}}\widetilde{\delta\mathbf{h}} + (\widetilde{\delta\mathbf{h}})^{\cdot}\right) && \text{(A.3)}\\
\delta(\tilde{\mathbf{v}}) - (\widetilde{\delta\mathbf{h}})^{\cdot} &= \tilde{\mathbf{v}}\widetilde{\delta\mathbf{h}} - \widetilde{\delta\mathbf{h}}\tilde{\mathbf{v}} && \text{(A.4)}\\
\delta(\tilde{\mathbf{v}}) - (\widetilde{\delta\mathbf{h}})^{\cdot} &= \left[\tilde{\mathbf{v}}, \widetilde{\delta\mathbf{h}}\right] && \text{(A.5)}
\end{aligned}$$

in which the $[\bullet,\bullet]$ operator is called the *Lie bracket* and is defined as

$$[\bullet,\bullet] \ : \ g \times g \to g, \ \left[\tilde{\mathbf{v}}, \widetilde{\delta\mathbf{h}}\right] \mapsto \tilde{\mathbf{v}}\widetilde{\delta\mathbf{h}} - \widetilde{\delta\mathbf{h}}\tilde{\mathbf{v}} \tag{A.6}$$

A.2 Exponential and logarithmic map

The exponential map $\exp_G(\bullet)$ maps an element $\widetilde{\mathbf{Q}}$ of the Lie algebra g to an element of the Lie group G, it is defined as

$$\exp_G(\bullet) \ : \ g \to G, \ \widetilde{\mathbf{Q}} \mapsto \exp_G(\widetilde{\mathbf{Q}}) \tag{A.7}$$

Mathematically, it is given by

$$\exp_G(\widetilde{\mathbf{Q}}) = \sum_{i=0}^{\infty} \frac{\widetilde{\mathbf{Q}}^i}{i!} \tag{A.8}$$

The inverse of the exponential map is called the logarithmic map $\log_G(\bullet)$. It maps an element $\mathbf{H}$ of the Lie group to an element of the Lie algebra and is defined as

$$\log_G(\bullet) \; : \; G \rightarrow g, \; \mathbf{H} \mapsto \log_G(\mathbf{H}) \tag{A.9}$$

and is given by

$$\log_G(\mathbf{H}) = \sum_{i=1}^{\infty} \frac{(\mathbf{I} - \mathbf{H})^i}{i} \tag{A.10}$$

The tangent operator is expressed as

$$\mathbf{T}(\mathbf{Q}) = \sum_{i=0}^{\infty} (-1)^i \frac{\hat{\mathbf{Q}}^i}{(i+1)!} \tag{A.11}$$

where the $\hat{\bullet}$ operator is the same as in Eq. (3.6).

A.2.1 Case of the $SO(3)$ group

Let us consider an element $\mathbf{R}$ of the special orthogonal group $SO(3)$ and a vector $\mathbf{v}_R \in \mathbb{R}^3$. The latter vector can be mapped to the lie algebra of $SO(3)$ using the $\tilde{\bullet}$ operator as $\tilde{\mathbf{v}}_R$. The exponential map $\exp_{SO(3)}(\bullet)$ in the case of the $SO(3)$ group is given by the Rodrigue's formula, as

$$\exp_{SO(3)}(\tilde{\mathbf{v}}_R) = \mathbf{I}_{3\times 3} + \alpha(\mathbf{v}_R)\tilde{\mathbf{v}}_R + \frac{\beta(\mathbf{v}_R)}{2}\tilde{\mathbf{v}}_R^2 \tag{A.12}$$

with

$$\alpha(\mathbf{v}_R) = \frac{\sin(\|\mathbf{v}_R\|)}{\|\mathbf{v}_R\|} \quad ; \quad \beta(\mathbf{v}_R) = 2\frac{1-\cos(\|\mathbf{v}_R\|)}{\|\mathbf{v}_R\|^2} \tag{A.13}$$

Notice that $\alpha(\mathbf{0}) = \beta(\mathbf{0}) = 1$. The logarithmic map $\log_{SO(3)}(\bullet)$ is given by

$$\log_{SO(3)}(\mathbf{R}) = \frac{\theta}{2\sin(\theta)}(\mathbf{R} - \mathbf{R}^T) \tag{A.14}$$

with $\theta = \arccos\left(\frac{1}{2}(\text{trace}(\mathbf{R}) - 1\right)$ and $-\pi < \theta < \pi$. One can note that in the case of the $SO(3)$ group, the $\tilde{\bullet}$ operator and the $\log_{SO(3)}(\bullet)$ are equivalent.

One can also define the tangent operators $\mathbf{T}_{SO(3)}(\bullet)$ as

$$\mathbf{T}_{SO(3)}(\mathbf{v}_R) = \mathbf{I}_{3\times 3} - \frac{\beta(\mathbf{v}_R)}{2}\tilde{\mathbf{v}}_R + \frac{1-\alpha(\mathbf{v}_R)}{\|\mathbf{v}_R\|^2}\tilde{\mathbf{v}}_R^2 \tag{A.15}$$

and its inverse as

$$\mathbf{T}^{-1}_{SO(3)}(\mathbf{v}_R) = \mathbf{I}_{3\times 3} + \frac{1}{2}\tilde{\mathbf{v}}_R + \frac{1-\gamma(\mathbf{v}_R)}{\|\mathbf{v}_R\|^2}\tilde{\mathbf{v}}_R^2 \tag{A.16}$$

with

$$\gamma(\mathbf{v}_R) = \frac{\|\mathbf{v}_R\|}{2}\cot(\frac{\|\mathbf{v}_R\|}{2}) \tag{A.17}$$

It can be noted that for small values of $\mathbf{v}_R$, i.e. $\|\mathbf{v}_R\| \ll 1$, the tangent operator $\mathbf{T}_{SO(3)} \approx \mathbf{I}_{3\times 3}$.

A.2.2 Case of the $SE(3)$ group

One now consider the case of an element $\mathbf{H}(\mathbf{R}, \mathbf{p})$ of the special Euclidean group $SE(3)$. A vector $\mathbf{v}(\mathbf{v}_R, \mathbf{v}_p) \in \mathbb{R}^6$ can be mapped to the Lie algebra of $SE(3)$ using the $\tilde{\bullet}$ operator as $\tilde{\mathbf{v}}$. The exponential map $\exp_{SE(3)}(\bullet)$ in this case is given by

$$\exp_{SE(3)}(\tilde{\mathbf{v}}) = \begin{bmatrix} \exp_{SO(3)}(\tilde{\mathbf{v}}_R) & \mathbf{T}^T_{SO(3)}(\mathbf{v}_R)\mathbf{v}_p \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix} \tag{A.18}$$

and the logarithmic map $\log_{SE(3)}(\bullet)$ is given by

$$\log_{SE(3)}(\mathbf{H}) = \begin{bmatrix} \tilde{\mathbf{v}}_R & \mathbf{T}^{-T}_{SO(3)}(\mathbf{v}_R)\mathbf{p} \\ \mathbf{0}_{1\times 3} & 0 \end{bmatrix} \tag{A.19}$$

The tangent operators $\mathbf{T}_{SE(3)}(\bullet)$ in the $SE(3)$ group is given by

$$\mathbf{T}_{SE(3)}(\mathbf{v}) = \begin{bmatrix} \mathbf{T}_{SO(3)}(\mathbf{v}_R) & \mathbf{T}_{pR+}(\mathbf{v}_p, \mathbf{v}_R) \\ \mathbf{0}_{3\times 3} & \mathbf{T}_{SO(3)}(\mathbf{v}_R) \end{bmatrix} \tag{A.20}$$

where $\mathbf{T}_{pR+}(\mathbf{v}_p, \mathbf{v}_R)$ is given by

$$\begin{aligned}\mathbf{T}_{pR+}(\mathbf{v}_p, \mathbf{v}_R) = {} & \frac{-\beta(\mathbf{v}_R)}{2}\tilde{\mathbf{v}}_p + \frac{1-\alpha(\mathbf{v}_R)}{\|\mathbf{v}_R\|^2}\lceil \mathbf{v}_p, \mathbf{v}_R \rceil \\ & + \frac{\mathbf{v}_R^T\mathbf{v}_p}{\|\mathbf{v}_R\|^2}\left((\beta(\mathbf{v}_R) - \alpha(\mathbf{v}_R))\tilde{\mathbf{v}}_R + (\frac{\beta(\mathbf{v}_R)}{2} - \frac{3(1-\alpha(\mathbf{v}_R))}{\|\mathbf{v}_R\|^2})\tilde{\mathbf{v}}_R^2\right)\end{aligned} \tag{A.21}$$

where $\lceil \mathbf{v}_p, \mathbf{v}_R \rceil = \tilde{\mathbf{v}}_p\tilde{\mathbf{v}}_R + \tilde{\mathbf{v}}_R\tilde{\mathbf{v}}_p$. The inverse of this tangent operator is given by

$$\mathbf{T}_{SE(3)}^{-1}(\mathbf{v}) = \begin{bmatrix} \mathbf{T}_{SO(3)}^{-1}(\mathbf{v}_R) & \mathbf{T}_{pR-}(\mathbf{v}_p, \mathbf{v}_R) \\ \mathbf{0}_{3\times3} & \mathbf{T}_{SO(3)}^{-1}(\mathbf{v}_R) \end{bmatrix} \tag{A.22}$$

where $\mathbf{T}_{pR-}(\mathbf{v}_p, \mathbf{v}_R)$ is given by

$$\mathbf{T}_{pR-}(\mathbf{v}_p, \mathbf{v}_R) = \frac{1}{2}\tilde{\mathbf{v}}_p + \frac{1-\gamma(\mathbf{v}_R)}{\|\mathbf{v}_R\|^2}\lceil \mathbf{v}_p, \mathbf{v}_R \rceil + \frac{1/\beta(\mathbf{v}_R) + \gamma(\mathbf{v}_R) - 2}{\|\mathbf{v}_R\|^4}(\mathbf{v}_R^T\mathbf{v}_p)\tilde{\mathbf{v}}_R^2 \tag{A.23}$$

A.3 Double pendulum in minimal coordinates

The development of the equations for the under-actuated double pendulum described in Fig. 2.5 are given below.

The kinetic energy $\mathcal{K}$ of the system is expressed as

$$\mathcal{K} = ml^2\dot{q}_1^2 + \frac{1}{2}ml^2\dot{q}_2^2 + ml^2\dot{q}_1\dot{q}_2\cos(q_1 - q_2) \tag{A.24}$$

and the potential energy $\mathcal{V}$ of the spring is

$$\mathcal{V} = \frac{1}{2}k(q_2 - q_1)^2 \tag{A.25}$$

The non-linear equations of motion can be derived from the Lagrangian $\mathcal{L} = \mathcal{K} - \mathcal{V}$ as

$$\frac{d}{dt}\left(\frac{d\mathcal{L}}{d\dot{q}_i}\right) - \frac{d\mathcal{L}}{dq_i} = \mathbf{0} \tag{A.26}$$

with each derivation developed as

$$\frac{d\mathcal{L}}{dq_1} = -ml^2\dot{q}_1\dot{q}_2\sin(q_1 - q_2) + (q_2 - q_1) \tag{A.27}$$

$$\frac{d\mathcal{L}}{dq_2} = ml^2\dot{q}_1\dot{q}_2\sin(q_1 - q_2) - (q_2 - q_1) \tag{A.28}$$

$$\frac{d\mathcal{L}}{d\dot{q}_1} = 2ml^2\dot{q}_1 + ml^2\dot{q}_2\cos(q_1 - q_2) \tag{A.29}$$

$$\frac{d\mathcal{L}}{d\dot{q}_2} = 2ml^2\dot{q}_2 + ml^2\dot{q}_1\cos(q_1 - q_2) \tag{A.30}$$

$$\frac{d}{dt}\left(\frac{d\mathcal{L}}{d\dot{q}_1}\right) = 2ml^2\ddot{q}_1 + ml^2\left[\ddot{q}_2\cos(q_1 - q_2) + \dot{q}_2\left[\sin(q_1 - q_2)\dot{q}_2 - \sin(q_1 - q_2)\dot{q}_1\right]\right] \tag{A.31}$$

$$\frac{d}{dt}\left(\frac{d\mathcal{L}}{d\dot{q}_2}\right) = ml^2\ddot{q}_2 + ml^2\left[\ddot{q}_1\cos(q_1 - q_2) + \dot{q}_1\left[\sin(q_1 - q_2)\dot{q}_2 - \sin(q_1 - q_2)\dot{q}_1\right]\right] \tag{A.32}$$

If one only considers an equilibrium around $(q_1 - q_2) \ll 1$, one can approximate $\cos(q_1 - q_2)$ and $\sin(q_1 - q_2)$ to 1 and 0 respectively and the linearized form of the equations of motion of this under-actuated pendulum are

$$2ml^2\ddot{q}_1 + ml^2\ddot{q}_2 - k(q_2 - q_1) = u \tag{A.33}$$

$$ml^2\ddot{q}_1 + ml^2\ddot{q}_2 + k(q_2 - q_1) = 0 \tag{A.34}$$

$$q_2 = y_{\text{presc}}(t) \tag{A.35}$$

As already presented in Chapter 2, the feedforward input torque u_{FF} can be expressed as

$$u_{\text{FF}} = 3ml^2\ddot{q}_1 + 2ml^2\ddot{y}_{\text{presc}} \tag{A.36}$$

where the dynamics of $\ddot{q}_1$ still needs to be solved.

Bibliography

[1] P.-A. Absil, R. Mahony, and R. Sepulchre. *Optimization algorithms on Matrix Manifolds*. Princeton University Press, 2008.

[2] Y. Aoustin and A. Formal'sky. On the feedforward torques and reference trajectory for flexible two-link arm. *Multibody System Dynamics*, 3:241–265, 1999.

[3] H. Asada, Z. Ma, and H. Tokumaru. Trajectory control and inverse dynamics of flexible robot arms. *Journal of Dynamic Systems, Measurement and Control*, 112(5):177–185, 1990.

[4] G. J. Bastos. *Contribution to the Inverse Dynamics of Flexible Manipulators*. PhD thesis, University of Liège, 2013.

[5] G. J. Bastos, R. Seifried, and O. Brüls. Inverse dynamics of underactuated multibody systems using a DAE optimal control approach. *In Proceedings of the ECCOMAS Thematic Conference (Multibody Dynamics)*, 2011. Brussels, Belgium.

[6] G. J. Bastos, R. Seifried, and O. Brüls. Inverse dynamics of serial and parallel underactuated multibody systems using a DAE optimal control approach. *Multibody System Dynamics*, 30:359–376, 2013.

[7] G. J. Bastos, R. Seifried, and O. Brüls. Analysis of stable model inversion methods for constrained underactuated mechanical systems. *Mechanism and Machine Theory*, 111:99–117, 2017.

[8] O. Bauchau. *Flexible multibody dynamics*. Springer, 2011.

[9] E. Bayo. A finite-element approach to control the end-point motion of a single link flexible robot. *Journal of Robotic Systems*, 1987.

[10] E. Bayo, P. Papadopoulos, J. Stubbe, and M. A. Serna. Inverse dynamics and kinematics of multi-link elastic robots: An iterative frequency domain approach. *The International Journal of Robotics Research*, 1989.

[11] M. Benosman and G. Le Vey. End-effector motion planning for one-link flexible robot. In *Proceedings of the 6th IFAC Symposium on Robot Control*, volume 33, pages 315–320, 2000.

[12] M. Benosman and G. Le Vey. Control of flexible manipulators: A survey. *Robotica*, 22(5):533–545, 2004.

[13] J. T. Betts. *Practical Method for Optimal Control and Estimation Using Nonlinear Programming.* Advances in Desgin and Control. SIAM, 2010.

[14] W. Blajer. Dynamics and control of mechanical systems in partly specified motion. *Journal of the Franklin Institute*, 334(3):407–426, 1997.

[15] 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.

[16] W. J. Book. Recursive Lagrangian dynamics of flexible manipulator arms via transformation matrices. In *Proceedings of the 2nd IFAC Symposium on Computer Aided Design of Multivariable Technological systems*, volume 16, pages 5–17, 1983.

[17] W. J. Book. Controlled motion in an elastic world. *Journal of Dynamic Systems, Measurement, and Control*, 115:252–261, 1993.

[18] C. L. Bottasso and A. Croce. Optimal control of multibody systems using an energy preserving direct transcription method. *Multibody System Dynamics*, 12:17–45, 2004.

[19] C. L. Bottasso, A. Croce, L. Ghezzi, and P. Faure. On the solution of inverse dynamics and trajectory optimization problems for multibody systems. *Multibody System Dynamics*, 11:1–22, 2004.

[20] O. Brüls, V. Acary, and A. Cardona. Simultaneous enforcement of constraints at position and velocity levels in the nonsmooth generalized-*alpha* scheme. *Computer Methods in Applied Mechanics and Engineering*, 281(1):131–161, 2014.

[21] O. Brüls, M. Arnold, and A. Cardona. Two Lie group formulations for dynamic multibody systems with large rotations. *In Proceedings of the ASME 2011 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference*, 2011. Washington D.C., U.S.

[22] O. Brüls, G. J. Bastos, and R. Seifried. A stable inversion method for feedforward control of constrained flexible multibody systems. *Journal of Computational and Nonlinear Dynamics*, 9, 2014.

[23] O. Brüls, A. Cardona, and M. Arnold. Lie group generalized-α time integration of constrained flexible multibody systems. *Mechanism and Machine Theory*, 48:121–137, 2012.

[24] O. Brüls, A. Cardona, and M. Geradin. *Simulation Techniques for Applied Dynamics*, volume 507 of *CISM International Centre for Mechanical Sciences*,

chapter Modelling, simulation and control of flexible multibody systems, pages 21–74. Springer, 2008.

[25] O. Brüls, P. Duysinx, and J.-C. Golinval. The global modal parameterization for non-linear model-order reduction in flexible multibody dynamics. *International journal for numerical methods in engineering*, 69:948–977, 2006.

[26] M. Burkhardt, R. Seifried, and P. Eberhard. Experimental studies of control concepts for a parallel manipulator with flexible links. In *Proceedings of the third Joint International Conference on Multibody System Dynamics*, 2014.

[27] S. L. Campbell. High-index differential algebraic equations. *Mechanics of Structures and Machines*, 23(2):199–222, 1995.

[28] R. Cannon and E. Schmitz. Initial experiments on the end-point control of a flexible one-link robot. *The International Journal of Robotics Research*, 3(3):62–75, 1984.

[29] D. Chen and B. Paden. Stable inversion of nonlinear non-minimum phase systems. *International Journal of Control*, 1996.

[30] Z. Chu and J. Cui. Experiment on vibration control of a two-link flexible manipulator using an input shaper and adaptive positive position feedback. *Advances in Mechanical Engineering*, 7(10):1–13, 2015.

[31] W. K. Chung, L.-C. Fu, and T. Kröger. *Springer Handbook of Robotics*, chapter 8. Motion Control, pages 163–194. Springer, 2016.

[32] R. Clavel. *Conception d'un robot parallele rapide a 4 degres de liberte*. PhD thesis, Ecole Polytechnique Federale Lausanne, 1991.

[33] A. De Luca. Feedforward/feedback laws for the control of flexible robots. *In Proceedings of the IEEE International Conference on Robotics & Automation*, 2000.

[34] A. De Luca and W. J. Book. *Springer Handbook of Robotics*, chapter 11. Robots with flexible elements, pages 243–282. Springer, 2016.

[35] A. De Luca and G. Di Giovanni. Rest-to-rest motion of a one-link flexible arm. In *IEEE/ASME International Conference on Advanced Intelligent Mechatronics Proceedings*, 2001.

[36] A. De Luca, P. Lucibello, and G. Ulivi. Inversion techniques for trajectory control of flexible robot arms. *Journal of Robotic Systems*, 6(4), 1989.

[37] A. De Luca and B. Siciliano. Inversion-based nonlinear control of robot arms with flexible links. *Journal of Guidance, Control, and Dynamics*, 1993.

[38] J. Denavit and R. Hartenberg. A kinematic notation for lower pair mechanisms based on matrices. *Journal of Applied Mechanics*, 1955.

[39] S. Devasia. Approximate stable inversion for nonlinear systems with nonhyperbolic internal dynamics. In *Proceedings of the 36th IEEE Conference on Decision and Control*, 1997.

[40] S. Devasia, D. Chen, and B. Paden. Nonlinear inversion-based output tracking. *IEEE Transactions on Automatic Control*, 41(7), 1996.

[41] P. E. Dupont. Friction modeling in dynamic robot simulation. In *Proceedings of the IEEE International Conference on Robotics and Automation*, 1990.

[42] S. K. Dwivedy and P. Eberhard. Dynamic analysis of flexible manipulators, a literature review. *Mechanism and Machine Theory*, 41:749–777, 2006.

[43] M. Farid and S. A. Lukasiewicz. Dynamic modeling of spatial manipulators with flexible links and joints. *Computers and Structures*, 75, 2000.

[44] M. Fliess, J. Levine, P. Martin, and P. Rouchon. On differentially flat nonlinear systems. In *Proceedings of the 2nd IFAC Symposium on Nonlinear Control Systems Design*, 1992.

[45] M. Fliess, J. Levine, P. Martin, and P. Rouchon. Flatness and defect of nonlinear systems: Introductory theory and examples. *International Journal of Control*, 1995.

[46] R. Franke, J. Malzahn, T. Nierobisch, F. Hoffmann, and T. Bertram. Vibration control of a multi-link flexible robot arm with fiber-bragg-grating sensors. *In Proceedings of IEEE International Conference on Robotics and Automation*, 2009.

[47] A. Fumagalli and P. Masarati. Real-time inverse dynamics control of parallel manipulators using general-purpose multibody software. *Multibody System Dynamics*, 22:47–68, 2009.

[48] L. Ganovski. *Modeling, Simulation and Control of Redundantly Actuated Parallel Manipulators*. PhD thesis, Université Catholique de Louvain, 2007.

[49] L. Ganovski, P. Fisette, and J.-C. Samin. Piecewise overactuation of parallel mechanisms following singular trajectories: Modeling, simulation and control. *Multibody System Dynamics*, 12:317–343, 2004.

[50] H. Gattringer, A. Müller, K. Springer, and M. Jörgl. An efficient method for the dynamical modeling of serial elastic link/joint robots. In *Computer Aided Systems Theory - EUROCAST 2015*, volume 9520 of *Lecture Notes in Computer Science*, page 689. Springer International Publishing, 2015.

[51] M. Geradin and A. Cardona. *Flexible multibody dynamics : a finite element approach*. John Wiley, 2001.

[52] M. C. Good, L. M. Sweet, and K. L. Strobel. Dynamic models for control system design of integrated robot and drive systems. *Journal of Dynamic Systems, Measurement and Control*, 107:53–59, 1985.

[53] K. Graichen, V. Hagenmeyer, and M. Zeitz. A new approach to inversion-based feedforward control design for nonlinear systems. *Automatica*, 41:2033–2041, 2005.

[54] S. Grazioso, G. Di Gironimo, and B. Siciliano. A geometrically exact model for soft continuum robots: The finite element deformation space formulation. *Soft robotics*, 6, 2018.

[55] S. Grazioso, V. Sonneville, G. Di Gironimo, O. Bauchau, and B. Siciliano. A nonlinear finite element formalism for modelling flexible and soft manipulators. In *Proceedings of the 2016 IEEE International Conference on Simulation, Modeling, and Programming for Autonomous Robots*, 2016.

[56] A. Green and J. Sasiadek. Dynamics and trajectory tracking control of a two-link robot manipulator. *Journal of Vibration and Control*, 2004.

[57] M. Hermle and P. Eberhard. Control and parameter optimization of flexible robots. *Mechanics of Structures and Machines*, 28(2):137–168, 2000.

[58] R. Hirschorn. Invertibility of nonlinear control systems. *SIAM Journal on Control and Optimization*, 17(2):289–297, 1979.

[59] S. Hoshyari, H. Xu, E. Knoop, S. Coros, and M. Bächer. Vibration-minimizing motion retargeting for robotic characters. *ACM Transactions on Graphics*, 38(4), 2019.

[60] M. T. Hussein and D. Söffker. State variables estimation of flexible link robot using vision sensor data. In *Proceedings of the 7th Vienna International Conference on Mathematical Modelling*, volume 45, 2012.

[61] H. N. Huynh. *Robotic machining: Development and validation of a numerical model of robotic milling to optimise the cutting parameters.* PhD thesis, University of Mons, 2019.

[62] A. Isidori. *Nonlinear Control System.* Communication and Control Engineering Series. Springer, 1995.

[63] M. Kelly. An introduction to trajectory optimization: How to do your own direct collocation. *SIAM Review*, 59(4):849–904, 2017.

[64] S. Krenk. Dispertion-corrected explicit integration of the wave equation. *Computer methods in applied mechanics and engineering*, 191:975–987, 2001.

[65] D.-S. Kwon and W. J. Book. A time-domain inverse dynamic tracking control of a single link flexible manipulator. *Journal of Dynamic Systems, Measurement and Control*, 116:193–200, 1994.

[66] J. D. Lee. Application of optimal control theory to flexible robotic manipulations. *Robotics and Computer Integrated Manufacturing*, 7, 1990.

[67] A. Lismonde, V. Sonneville, and O. Brüls. Trajectory planning of soft link robots with improved intrinsic safety. *In Proceedings of the 20th World Congress of the International Federation of Automatic Control*, 2017.

[68] K. Lochan, B. Roy, and B. Subudhi. A review on two-link flexible manipulators. *Annual Reviews in Control*, 2016.

[69] K. Lochan, B. Roy, and B. Subudhi. Smc controlled chaotic trajectory tracking of two-link flexible manipulator with pid sliding surface. In *Proceedings of the 4th IFAC Conference on Advances in Control and Optimization of Dynamical Systems*, 2016.

[70] K. M. Lynch and F. C. Park. *Modern robotics : mechanics, planning, and control.* Cambridge University Press, 2017.

[71] D. P. Magee and W. J. Book. The application of input shaping to a system with varying parameters. In *Proceedings 1992 Japan-USA Symposium on Flexible Automation*, 1992.

[72] J. Malzahn, M. Ruderman, A. S. Phung, F. Hoffmann., and T. Bertram. Input shaping and strain gauge feedback vibration control of an elastic robotic arm. *In Proceedings of IEEE Conference on Control and Fault Tolerant Systems*, 2010.

[73] S. Manara, M. Gabiccini, A. Artoni, and M. Diehl. On the integration of singularity-free representations of SO(3) for direct optimal control. *Nonlinear Dynamics*, 90(2):1223–1241, 2017.

[74] J. Martins, M. A. Botto, and J. Sa Da Costa. Modeling of flexible beams for robotic manipulators. *Multibody System Dynamics*, 7:79–100, 2002.

[75] J.-P. Merlet. *Parallel Robots.* Solid mechanics and its applications. Springer, second edition edition, 2006.

[76] D. K. Miu. Physical interpretation of transfer function zeros for simple control systems with mechanical flexibilities. *Journal of Dynamic Systems, Measurement and Control*, 113:419–424, 1991.

[77] M. Moallem, R. V. Patel, and K. Khorasani. *Flexible-link Robot Manipulators: Control techniques and Structural Design*, volume 257 of *Lecture notes in Control and Information Sciences.* Springer, 2000.

[78] S. Moberg. *Modeling and Control of Flexible Manipulators.* PhD thesis, Linköping University, 2010.

[79] S. Moberg and S. Hanssen. Inverse dynamics of flexible manipulators. *In Proceedings of the Multibody dynamics, ECCOMAS Thematic Conference*, 2009.

[80] Z. Mohamed, J. Martins, M. O. Tokhi, J. Sa da Costa, and M. A. Botto. Vibration control of a very flexible manipulator system. *Control Engineering Practice*, (13):267–277, 2005.

[81] Z. Mohamed and M. O. Tokhi. Command shaping techniques for vibration control of a flexible robot manipulator. *Mechatronics*, 14:69–90, 2004.

[82] M. Morlock, M. Burkhardt, and R. Seifried. Control of vibrations for a parallel manipulator with flexible links - concepts and experimental results. *Journal of Physics: Conference Series MOVIC2016 & RASD2016*, 744, 2016.

[83] H. Moulin and E. Bayo. On the accuracy of end-point trajectory tracking for flexible arms by noncausal inverse dynamic solutions. *Journal of Dynamic Systems, Measurement, and Control*, 1991.

[84] R. M. Murray, Z. Li, and S. S. Sastry. *A mathematical introduction to robotic manipulation*. CRC Press, 1994.

[85] S. Nicosia, F. Nicolo, and D. Lentini. Dynamical control of industrial robots with elastic and dissipative joints. In *Proceedings of the 8th IFAC World Congress on Control Science and Technology*, 1981.

[86] S. Nicosia, P. Tomei, and A. Tornambe. Nonlinear observers for flexible robots. In *Proceedings of the 11th IFAC World Congress on Automatic Control*, volume 23, 1990.

[87] C. Oakley and R. Cannon. Theory and experiments in selecting mode shapes for two-link flexible manipulators. In V. Hayward and O. Khatib, editors, *Experimental Robotics I. Lecture Notes in Control and Information Sciences*, volume 139. Springer, 1990.

[88] S. Otto and R. Seifried. *Applications of Differential-Algebraic Equations: Examples and Benchmarks*, chapter Open-Loop Control of Underactuated Mechanical Systems Using Servo-Constraints: Analysis and Some Examples, pages 81–122. Springer, 2019.

[89] M. Pala Prasad Reddy, S. K. V., and J. Jacob. Precise non linear modeling of flexible link flexible joint manipulator. *International Review on Modelling and Simulations*, 5(3):1369–1374, 2012.

[90] I. Payo, F. Ramos, O. D. Cortazar, and V. Feliu. Experimental validation of nonlinear dynamic models for single-link very flexible arms. In *Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference 2005*, 2005.

[91] E. Pereira, J. Becedas, I. Payo, F. Ramos, and V. Feliu. Control of flexible manipulators. theory and practice. *Robot Manipulators Trends and Development*, 2010.

[92] B. K. Post and W. J. Book. A robust nonlinear observation strategy for the control of flexible manipulators. In *IEEE International Conference on Robotics and Automation*, 2011.

[93] G. A. Pratt and M. M. Williamson. Series elastic actuators. In *Proceedings of the 1995 IEEE/RSJ International Conference on Intelligent Robots and Systems. Part 3 (of 3)*, 1995.

[94] A. Preumont. *Vibration Control of Active Structures, An Introduction 3rd edition*. Springer, 2011.

[95] S. Sastry. *Nonlinear Systems: Analysis, Stability and Control*, volume 10 of *Interdisciplinary Applied Mechanics*. Springer, 1999.

[96] M. Sayahkarajy, Z. Mohamed, and A. A. M. Faudzi. Review of modelling and control of flexible-link manipulators. In *Proceedings of the Institution of Mechanical Engineers Part I Journal of Systems and Control Engineering*, volume 230, pages 861–873, 2016.

[97] B. Scaglioni, L. Bascetta, M. Baur, and G. Ferretti. Closed-form control oriented model of highly flexible manipulators. *Applied Mathematical Modelling*, 52:174–185, 2017.

[98] R. Seifried. *Dynamics of Underactuated Multibody Systems: Modeling, Control and Optimal Design*. Solid Mechanics and its Applications. Springer, 2014.

[99] R. Seifried and W. Blajer. Analysis of servo-constraint problems for underactuated multibody systems. *Mechanical Sciences*, 4:113–129, 2013.

[100] R. Seifried and P. Eberhard. Design of feed-forward control for underactuated multibody systems with kinematic redundancy. *Motion and Vibration Control: Selected papers from MOVIC 2008*, 2009.

[101] J. M. Selig. *Geometric fundamentals of Robotics*. Monographs in Computer Science. Springer, second edition, 2005.

[102] A. Shabana. *Dynamics of Multibody Systems*. Cambridge University Press, fourth edition edition, 2013.

[103] J. C. Simo and L. Vu-Quoc. On the dynamics of flexible beams under large overall motions—the plane case: Part i. *Journal of Applied Mechanics*, 53(4), 1986.

[104] N. Singer and W. P. Seering. Preshaping command inputs to reduce system vibration. *Journal of Dynamic Systems, Measurement and Control*, 112:76–82, 1990.

[105] J.-J. E. Slotine and W. Li. *Applied Nonlinear Control*. Prentice Hall, 1991.

[106] J. F. P. Solis, G. S. Navarro, and R. C. Linares. Modeling and tip position control of a flexible link robot: Experimental results. *Computacion y Sistemas*, 12(4):421–435, 2009.

[107] V. Sonneville. *A geometric local frame approach for flexible multibody systems.* PhD thesis, University of Liège, 2015.

[108] V. Sonneville and O. Brüls. A formulation on the special Euclidean group for dynamic analysis of multibody systems. *Journal of Computational and Nonlinear Dynamics*, 9, 2014.

[109] V. Sonneville, A. Cardona, and O. Brüls. Geometrically exact beam finite element formulated on the special euclidean group SE(3). *Computer Methods in Applied Mechanics & Engineering*, 268:451–474, 2014.

[110] M. W. Spong. Modeling and control of elastic joint robots. *Journal of Dynamic Systems, Measurement, and Control*, 109(4):310–319, 1987.

[111] P. Staufer and H. Gattringer. State estimation on flexible robots using accelerometers and angular rate sensors. *Mechatronics*, 22(8):1043 – 1049, 2012.

[112] D. G. 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.

[113] R. J. Theodore and A. Ghosal. Comparison of the assumed modes and finite element models for flexible multi-link manipulators. *The International Journal of Robotics Research*, 14(2):91–111, 1995.

[114] E. Tromme. *Structural optimization of flexible components within a multibody dynamics approach.* PhD thesis, University of Liège, 2015.

[115] E. Tromme, A. Held, P. Duysinx, and O. Brüls. System-based approaches for structural optimization of flexible mechanisms. *Archives of Computational Methods in Engineering*, 25, 2017.

[116] M. J. Van Nieuwstadt and R. M. Murray. Approximate trajectory generation for differentially flat systems with zero dynamics. In *Procedings of the 34th Conference on Decision and Control*, 1995.

[117] M. J. Van Nieuwstadt and R. M. Murray. Real time trajectory generation for differentially flat systems. *International Journal of Robust and Nonlinear Control*, 8(11), 1998.

[118] R. Waiboer, R. Aarts, and B. Jonker. Velocity dependence of joint friction in robotic manipulators with gear transmissions. In *Proceedings of the ECCOMAS Thematic conference on Multibody Dynamics*, 2005.

[119] Y. Wang, Y. CAO, and H. XIA. Optimized continuous non-singular terminal sliding mode control of uncertain flexible manipulators. In *Proceedings of the 34th Chinese Control Conference*, page 3392–3397, 2015.

[120] T. M. Wasfy and A. K. Noor. Computational strategies for flexible multibody systems. *Applied Mechanics Reviews*, 56(6):553–613, 2003.

[121] Y. Yang. *Numerical methods for the inverse dynamics simulation of underactuated mechanical systems.* PhD thesis, Karlsruher Institut fur Technologie (KIT), 2016.