ALMA MATER STUDIORUM - UNIVERSITÀ DI BOLOGNA

DOTTORATO DI RICERCA

in

MECCANICA E SCIENZE AVANZATE DELL'INGEGNERIA

Settore concorsuale di afferenza:	09/A2
Settore scientifico disciplinare:	ING-IND/13
Ciclo:	33°

DYNAMICS OF UNDERACTUATED CABLE-DRIVEN PARALLEL ROBOTS

Presentata da:	Edoardo Idà
Supervisore:	Prof. Ing. Marco Carricato
Coordinatore scuola di dottorato:	Prof. Ing. Marco Carricato

Esame Finale anno 2021

ALMA MATER STUDIORUM - UNIVERSITÀ DI BOLOGNA

DYNAMICS OF UNDERACTUATED CABLE-DRIVEN PARALLEL ROBOTS

A dissertation submitted in partial fulfillment
of the requirements for the degree of Doctor of Philosophy
in Mechanics and Advanced Engineering Sciences

Edoardo Idà

Bologna, Italy,
January 9, 2021

To my mom, dad, and Vale.
I am most grateful for your
unlimited love and support.

Abstract

The crane is certainly one of the most ancient devices, still used today for the lifting and transportation of heavy objects. A payload is attached to one cable and displaced by adjusting the cable's length and, possibly, the rotation or translation of some additional crane components. Even though this device is underactuated by design, that is, there is no set of actuation parameters that can completely define the object position and orientation from a purely geometric point of view, it has been and it still is widely used, mainly when the ability to move bulky objects around large areas is of primary importance.

The last three decades showed an evolution of the simple overhead crane design towards faster, safer, and more controllable multi-cable machines: this evolution led to so-called Cable-Driven Parallel Robots (*CDPRs* in short). An end-effector (*EE* in short) is constrained and moved by a group of cables, whose number is typically larger than the degrees of freedom of the *EE*: since cables can only exert tensile forces, the presence of redundant actuators allows cables to pull against each other, thus maintaining the *EE* stable. There are, though, intermediate solutions between the simple crane and redundantly-actuated *CDPRs*: these solutions form the family of underactuated *CDPRs*, which employs less cables than the degrees of freedom of the *EE*. The use of underactuated *CDPRs*, equipped with a limited number of cables, is justified in several applications, in which the task to be performed requires a limited number of controlled freedoms or a limitation of mobility is acceptable in order to enhance accessibility, decrease complexity, and ultimately decrease cost. On the other hand, the lack of full constraints on the *EE* allows its motion even when actuators are locked. If the *EE* is not in static equilibrium when the actuators cease to move, it exhibits undesired (and possibly dangerous) oscillatory motions.

This thesis focuses on the dynamics of underactuated *CDPRs*, including various aspects of robotic theory and practice, such as workspace computation, parameter identification, and trajectory planning. After a brief introduction to *CDPRs*, *UACDPR* kinematic and dynamic models are analyzed, under the relevant assumption of inextensible cables. The *free* oscillatory motion of the *EE* (namely, its capability to oscillate when all actuators are locked), which is a unique feature of underactuated mechanisms, is studied in detail, from both a kinematic and a dynamic perspective. The free (small) oscillations of the *EE* around equilibria are proved to be *harmonic* and the corresponding *natural oscillation frequencies* are analytically computed. The free motion plays a fundamental role in several topics that are treated in the remainder of the thesis. *UACDPR workspace* computation and analysis are then performed. A new performance index is proposed for the analysis of the influence of actuator errors on cable tensions around equilibrium configurations, and a new type of workspace, called *tension-error-insensitive*, is defined as the set of poses that a *UACDPR EE* can statically attain even in presence of actuation errors, while preserving tensions between assigned (positive) bounds. *EE* free oscillations are then employed to conceive a novel

procedure aimed at identifying the *EE inertial parameters*. A significant feature of this approach is that it does not require the use of force or torque measurements. Moreover, a *self-calibration* procedure for the experimental determination of *UACDPR* initial cable lengths is developed, which consequently enables the robot to automatically infer the *EE initial pose* at machine start-up. Lastly, *trajectory planning* of *UACDPRs* is investigated. Two alternative methods are proposed, which aim at (i) reducing *EE* oscillations even when model parameters are uncertain or (ii) eliminate *EE* oscillations in case model parameters are perfectly known. *EE* oscillations are reduced in real-time by dynamically scaling a nominal trajectory and filtering it with an *input shaper*, whereas they can be eliminated if an off-line trajectory is computed that accounts for the system internal dynamics.

Contents

Chapter 1

Introduction

In the robotic landscape, parallel architecture mechanisms have been employed for many years. A parallel robot can be loosely described as a platform, i.e. the end-effector (*EE* in short) of the machine, moved around by a set of kinematic chains, the so-called legs of the robot. The typical advantages of such an architecture with respect to the serial (or anthropomorphic) one are:

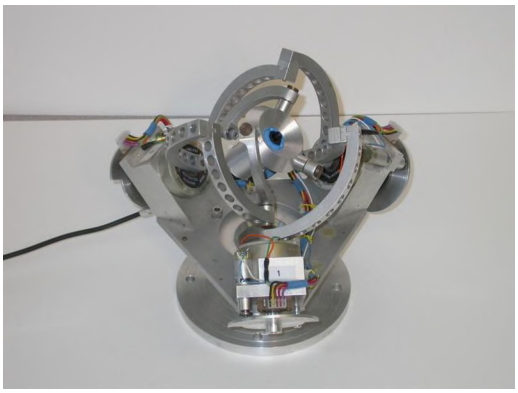
- high payload capability;
- modular design;
- remarkable accuracy and repeatability;

Unfortunately, the presence of numerous possibly-complex legs usually restrict the workspace of such manipulators, thus limiting their applications. In fact, sturdy legs (Fig. 1.1a) or complex kinematic chains (Fig. 1.1b) may interfeare with each other, and actuators limited strokes (Fig. 1.1a) represent a technological limit to dexterity and workspace dimensions.

1.1 Cable-Driven Parallel Robots

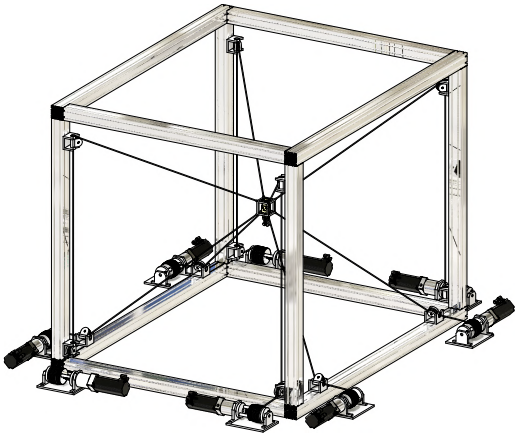
Cable-Driven Parallel Robots (*CDPRs*) form a class of parallel robots, introduced in the 80' [2,3], which employs cables in place of rigid-body extensible legs in order to control

(a) A Gough-Stewart platfrom (Source: Wikipedia)

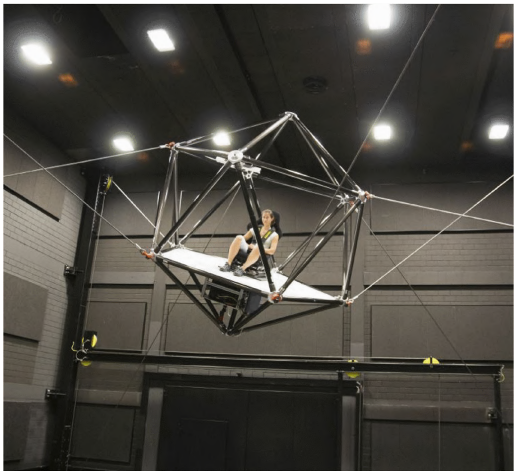


(b) The agile-eye [1]

Figure 1.1: Two types of parallel robots



(a) *CAD* model of a redundantly actuated *CDPR*



(b) The CableRobot simulator presented in [7]

Figure 1.2: *CDPR* with more cables than degrees of freedom.

the *EE* pose (Fig. 1.2a). Despite the fact that cables can only provide tensile force, that is, they can pull but not push, they present specific advantages over traditional parallel robots:

- flexible cables can be coiled and uncoiled onto motorized winches, thus the robot extensible legs can potentially be very long and its workspace remarkably large [4];
- cable mass is often practically negligible, thus increasing the robot dynamic performance [5];
- kinematic-chain elements, that is, winches and pulleys, can easily be rearranged, thus machine reconfigurability is enhanced [6].

Since guaranteeing positive tensions in cables is a necessary requisite to control the pose of the robot, the unilateral constraints imposed by cables complicate the control of the system.

An historical classification of *CDPRs* focuses on the relationship between the number n of cables which control the *EE*, and the degrees of freedom n_d (*DoFs* in short) of the latter, from a constrain-ability perspective [8]:

- if $n \leq n_d \leq 6$ the robot is defined underconstrained, since its *EE* cannot withstand an arbitrary external wrench with all the cables being under tension, or, conversely, cables cannot be under tension without a counterbalancing external load; in fact, these types of manipulators need at least an external action (i.e. gravity) to keep cables taut;
- if $n = n_d + 1$ the robot is defined completely-constrained, since cables are sufficient in number in order to effectively tension each other even without external counterbalancing loads;
- if $n > n_d + 1$ the robot is defined redundantly-constrained, since the number of cables is larger than the minimum number needed for constraining the *EE*; the presence of additional cables allows for the optimization of *CDPR* performance.

Most studies on *CDPRs* focused on manipulators with $n \geq n_d + 1$ (Fig. 1.2a): indeed, if the number of cables is larger that the degrees of freedom (*DoFs*) of the *EE*, and cables surround the *EE*, cables can be controlled to both displace the *EE* and tension each other [9]. Even though practical applications of these machines may be limited (since the *EE* is surrounded by cables, its interaction with the environment is hindered), *CDPR* with $n \geq n_d + 1$ were proposed for several applications, such as motion systems for wind tunnels [10], motion simulation platforms [7] (see also Fig. 1.2b), or automated storage and retrieval systems [11], just to cite a few. This class of *CDPRs* attracted so many researches over the years that a dedicated monography on the subject was recently published [12]: please refer to this book for additional information on these manipulators.

One of the limitations of the classification proposed in [8] regards underconstrained and completely-constrained *CDPRs*, since it is in contradiction with standard nomenclature in both analytical mechanics and control theory. In fact, in strictly physical terms, the *EE* is underconstrained if it has more *DoFs* than the dimension of the constraint space spanned by passive and active constraints [13], and thus it preserves a residual mobility even in case actuators are locked. In this case, the *EE* configuration does not depend only on geometric constraints (i.e. the actuators displacements), but also on mechanical equilibrium, namely the external loads that the *EE* is acted upon. This is the main difference between underconstrained systems (as classically defined) and those that are fully constrained: for the former, kinematics and statics (or dynamics) are coupled, and cannot be solved independently [14, 15]. In this perspective, we can infer that, notwithstanding the classification in [8]:

- if $n = n_d$, when all cables are taut the manipulator is completely-constrained, since the *EE* configuration is fully determined by cable lengths, and the solution of the inverse dynamic (or static) problem allows a single set of cable tensions to be computed (under the rigid-body assumption);
- if $n > n_d$, when all cables are taut the robot is redundantly-constrained, and the inverse dynamic (or static) problem admits infinite solutions under the rigid-body assumption;
- for any $n \geq n_d$, if the number of taut cables is less than n_d, the *EE* is still underconstrained;
- if $n < n_d$, the robot is necessarily underconstrained;
- the ability to be underconstrained, fully-constrained or redundantly-constrained is an instantaneous property, rather than an architectural one, and it does not depend only on the available number of actuated cables, but rather on the number of cables that are instantaneously taut.

The above-mentioned limitations of the classification proposed in [8] often leads to a misconception regarding the kinematic properties of *CDPRs*, with manipulators with both $n = n_d$ and $n < n_d$ being considered kinematically deficient. In fact, this is not true:

- any *CDPR* with $n \geq n_d$ is kinematically defined when at least n_d cable are taut [16];

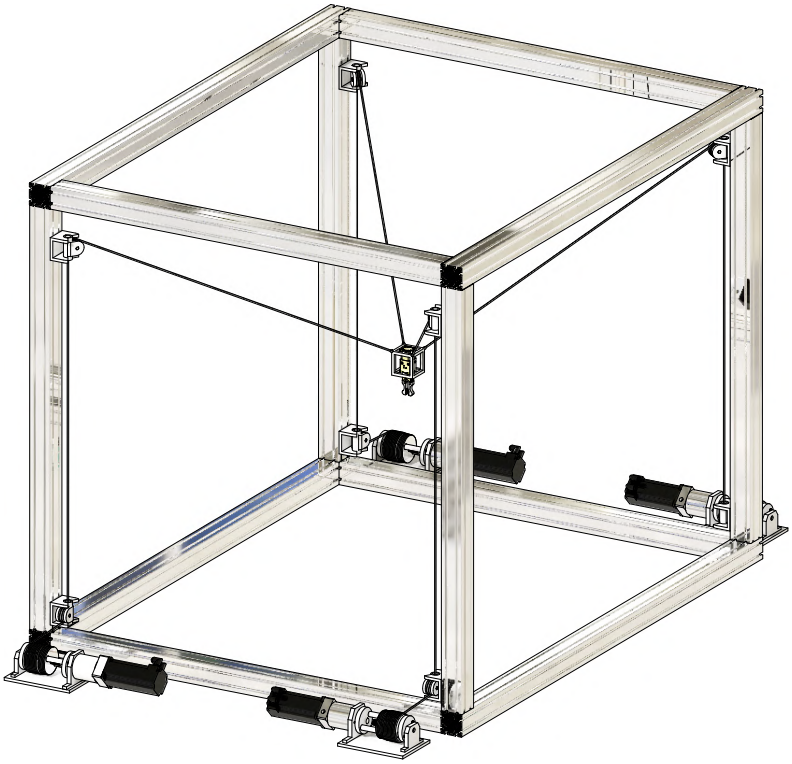


Figure 1.3: Underactuated suspended *CDPR*

- any *CDPR* with $n < n_d$ is kinematically deficient; any *CDPR* with $n \geq n_d$ where one or more cables become slack (for any reason) may become kinematically deficient.

Thus, kinematically-defined manipulators with $n \geq n_d$ are not to be confused or assimilated with kinematically-deficient robots with $n < n_d$. In the last decade, a growing literature regarding kinematically-defined *CDPR* with $n \geq n_d$ dealt with the study of dynamic trajectory planning of these systems, aiming at expanding their motion ability outside their *static workspace* [17–22].

A last remark is dedicated to a special class of systems that use cables arranged in a parallel fashion to constrain their end-effectors: they are usually assimilated to underconstrained *CDPRs*, but they are kinematically defined (or even redundant). It is the case of cable-driven systems whose structure can be discretely [23,24] or continuously [25] reconfigured, by moving the cables routing locations by means of trolley or drones.

Kinematically-deficient *CDPRs*, also called underactuated *CDPRs* (*UACDPRs* in short) since they have less actuators that their *EE DoFs*, are the focus of this thesis.

1.2 Underactuated Cable-Driven Parallel Robots

The use of *UACDPRs*, equipped with a limited number of cables which are usually routed towards the *EE* from an elevated position (suspended *UACDPRs*), is justified in several applications, in which the task to be performed requires a limited number of controlled freedoms or a limitation of mobility is acceptable in order to enhance accessibility, decrease complexity, and ultimately cost. A *CDPR* is underactuated if the number of actuated cables is less than the number of *DoFs* of the *EE* in the Cartesian space (Fig. 1.3). *UACDPR* are intrinsically underconstrained [15], since their *EE* is sub-

ject to less constaint actions than the number of its *DoFs*. As a consequence, the *EE* preserves some freedoms once the actuators are locked, a condition that is referred as *free motion* in the following.

For this class of manipulators, it is possible to prescribe a control law only for a subset of the generalized coordinates of the system. However, if an arbitrary motion is prescribed for a suitable subset of these coordinates, the constraint deficiency on the other freedoms may lead to the impossibility of bringing the system at rest in a prescribed time. In addition, the behavior of the system may not be stable, that is, oscillatory motion of the *EE* may arise.

Several challenges emerge in the analysis of underactuated systems, since most of the classical tools of robot analysis cannot be used in their traditional form. Nonetheless, the research dedicated to this class of manipulators is still limited.

The very first study on a *UACDPR* was conducted in [26], where the trajectory planning of a point-mass 2-*DoF* 1-cable system was analyzed. Planar 3-*DoF* 2-cable *UACDPRs* were considered in [27], where point-to-point motions were generated with parametric excitation of the cables, and in [28], where input-shaping was used to reduce *EE* oscillations during and after a trajectory execution. The latter strategy was then tested on spatial *UACDPRs* with 6-*DoF* actuated by 3 [29] and 4 [30] cables.

The geometrico-static modelling of *UACDPRs* was first addressed in [14]. Solutions to the forward [31–34] and inverse [35,36] problems, and the stability of these solutions [15], were addressed in detail.

UACDPRs were also studied from a control perspective, with the aim of stabilizing *EE* oscillations. Stabilizing controllers were synthesized for planar *UACDPRs* oscillating outside the robot work-plane [37], for spatial *UACDPRs* with 4 cables [38], and for purely planar *UACDPRs* [39].

Some efforts were also dedicated to the study of architecturally non-standard *UACDPRs*, which are often referred to as pendulum-like manipulators [40–42], since the cable-driven assembly, or the *EE* itself, move similarly to the well-known simple or double pendulum.

1.3 Thesis Goals and Outline

This thesis aims at analyzing *UACDPRs* dynamics, especially focusing on the cause of the unconstrained motion of the robot *EE*: the internal-dynamics of the system [43]. On the one hand, the internal dynamics poses control problems, since *EE* coordinates cannot be fully prescribed nor controlled, and undesirable platform oscillations may be excited during motion. On the other hand, internal dynamics can be interpreted as a second-order non-holonomic constraint on the robot coordinates, since it does not involve actuation or constraint actions in its formulation: this constraint can be exploited to devise efficient algorithms dedicated to *UACDPRs* for the solution of classical robotic problems, such as inertial-parameter identification. When this thesis started, its main goal was to study the trajectory planning of *UACDPRs*, with a strong focus on experimental validation of the results. By the time the first theoretical results were available for field-testing, the author realized that a lot of standard practical problems for completely or redundantly actuated manipulators had no straightforward nor readily-available solution for *UACDPRs*, and both experimentation and result analysis were severely complicated. Then, the focus of this thesis broadened to several practical issues of robotics applied to *UACDPRs*, such as worskpace characterization and

parameter identification.

Chapter 2 is dedicated to general kinematic and dynamic modelling of *UACDPRs*. The geometric constraints imposed by cables and swivel pulleys on the moving platform are analyzed, as well as their first- and second-order derivatives. The concepts of *EE free motion* and *controlled motion* are introduced and ground rules for their description are formally laid-out. Then, the *EE* dynamics is considered, and analytical methodology for internal dynamics derivation is proposed. In the end, static modelling, which is a special case of dynamics, is also considered. Two novel stiffness matrices, which have special meaning to *UACDPRs* and their analysis, are proposed.

Chapter 3 characterizes the static workspace of *UACDPRs*. This workspace is intrinsically a reachable one, since the *EE DoFs* cannot be fully controlled. A new performance index, called *maximum tension variation under a unit-norm cable displacement*, is proposed for the analysis of the influence of actuator errors in *UACDPR* cable tensions. Lastly, a new type of workspace, called *tension-error-insensitive*, is defined as the set of poses that a *UACDPR EE* can statically attain even in presence of actuation errors, while cable tensions remain within assigned (positive) bounds.

Chapter 4 explores how to model and/or experimentally determine some *UACDPR* parameters, namely its natural oscillation frequencies, its *EE* intertial parameters, and the cable lengths at machine start-up:

- natural oscillation frequencies are modelled as the solution of the eigenproblem arising from the linearization of the free-motion internal dynamics about equilibrium configurations; this model is validated by extensive experimental testing on 2-, 3-, and 4-cable robots;

- *EE* inertial parameters are experimentally identified according to a novel, internal-dynamics-based, identification methodology; the use of this model allows one to avoid any force or torque measurement for identification purposes; additionally, the use of free motion as identification exciting trajectory allows a limited number of kinematic measurements to be employed for identification, thus simplifying the experimental campaign;

- the estimation of initial cable lengths is performed by solving an extended (overdetermined) forward geometrico-static problem at each machine start-up; the data needed by the problem are recorded by following an automatic procedure, which allows cable lengths and swivel-pulley angle displacements to be recorded while maintaining cables under tension.

Chapter 5 is devoted to trajectory planning of *UACDPRs* controllable coordinates. The use of input-shaping and its combination with dynamic scaling is investigated in order to reduce *EE* oscillation in real-time with limited knowledge of robot geometric and dynamic parameters. Alternatively, the possibility to completely stop the *EE* after an arbitrary trajectory is considered by designing so-called rest-to-rest trajectories. This latter type of trajectories need to be computed off-line relying on a precise knowledge of robot parameters.

In the end, Chapter 6 draws conclusions and analyzes some issues that are still open.

Chapter 2

Modelling

This Chapter aims at introducing the mathematical models of a generic 6-*DoF* n-cable *UACDPR* moving in the Special Euclidean space of dimension 3 ($SE(3)$ in short). In this thesis, cables are modelled as straight line segments, massless and inextensible. They are coiled and uncoiled by servo-controlled winches, and their lengths are assumed to vary proportionally to actuator displacements. Moreover, cables are routed inside the workspace of the manipulator by means of swivel pulleys, whose geometry is considered in this work.

Section 2.1 presents *UACDPR* geometric modelling. *UACDPR* first and second order differential kinematic models are described in Section 2.2 also introducing the concept of *free* and *controlled* kinematics. Section 2.3 is dedicated to *UACDPR* dynamic models, more specifically to *EE* internal dynamics. The concept of *free* and *controlled* motions is applied to *EE* dynamics in order to underline some specific characteristics of underactuated manipulators.[1]

2.1 Geometric Modelling

A *CDPR* consists of a mobile platform coupled to the base by $n < 6$ cables, which can be coiled and uncoiled by motorized winches. In the following, $Oxyz$ is an inertial frame, whereas $Px'y'z'$ is a mobile frame attached to the moving platform, whose pose is described by the position vector $\mathbf{p}$ of P, and the rotation matrix $\mathbf{R}$ (Fig. 2.1). In this thesis, $\mathbf{R}$ is parametrized by a minimal set of orientation parameters, i.e. by tilt and torsion angles $\boldsymbol{\epsilon} = [\phi, \theta, \chi]^T$ [47], namely:

$$\mathbf{R}(\phi,\theta,\chi) = \mathbf{R}_a(\phi,\theta)\mathbf{R}_z(\chi), \qquad \mathbf{R}_a(\phi,\theta) = \mathbf{R}_z(\phi)\mathbf{R}_y(\theta)\mathbf{R}_z(-\phi) \tag{2.1}$$

where a is a line obtained by rotating the y-axis by ϕ around the z-axis, $\mathbf{R}_a(\phi,\theta)$ is a rotation θ about a, $\mathbf{R}_y(\cdot)$ and $\mathbf{R}_z(\cdot)$ are elementary rotation matrices about the y and z axes. Tilt and torsion angles often allow for an easy visualization of the orientation properties of the *EE*, even though a non-minimal set of orientation parameters would avoid representation singularities. A non-minimal set is not preferred in this work because, in the author opinion, it would unnecessarily complicate the mathematical model description. *EE* generalized coordinates are finally denoted by $\boldsymbol{\zeta} = \left[\mathbf{p}^T \, \boldsymbol{\epsilon}^T\right]^T$.

[1] The content of this Chapter was partly sumitted to *IEEE Transaction on Robotics* and was partly published in [44–46].

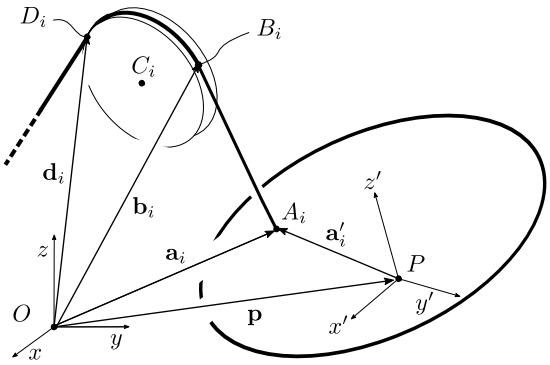


Figure 2.1: *CDPR* Geometric Model

Each cable is guided into the workspace by a swivel pulley of radius r_i and center C_i, mounted on an hinged support. The hinge axis, also called *swivel axis* in this work, is denoted as z_i, and it is tangent to the pulley in point D_i (Fig. 2.2a, 2.2b). The cable enters the pulley groove in D_i, exits from it at point B_i, and it is attached to the platform at point A_i. The coordinates of point D_i are constant in the inertial frame and denoted by $\mathbf{d}_i$, whereas the coordinates of A_i are constant in the mobile frame, and denoted by ${}^P\mathbf{a}'_i$; thus, the coordinates of A_i in the inertial frame are computed as:

$$\mathbf{a}_i = \mathbf{p} + \mathbf{a}'_i = \mathbf{p} + \mathbf{R}\,{}^P\mathbf{a}'_i, \qquad \mathbf{a}'_i = \mathbf{R}\,{}^P\mathbf{a}'_i \tag{2.2}$$

The position of point B_i depends on the pulley geometrical model [45, 48], which is thus detailed hereafter. It is convenient, for each cable, to define an additional fixed reference frame $D_i x_i y_i z_i$ centered in D_i, whose (constant) orientation is described in $Oxyz$ by unit vectors directed along x_i, y_i, z_i axes, namely $\mathbf{i}_i, \mathbf{j}_i, \mathbf{k}_i$ respectively (Fig. 2.2). The line through A_i and D_i and the swivel axis z_i are assumed to be co-planar and, as a consequence, $\boldsymbol{\varrho}_i = \mathbf{a}_i - \mathbf{d}_i$ and $\mathbf{k}_i$ define the *pulley plane*.

If $\sigma_i \in [-\pi, \pi]$, the *swivel angle*, is the angle between the coordinate plane $x_i z_i$ and the pulley plane (Fig. 2.2a), the unit vector $\mathbf{w}_i$ normal to the latter is:

$$\mathbf{w}_i = -\sin(\sigma_i)\,\mathbf{i}_i + \cos(\sigma_i)\,\mathbf{j}_i \tag{2.3}$$

Additionally, the unit vector $\mathbf{u}_i$ directed as $C_i - D_i$ can be found as $\mathbf{u}_i = \mathbf{w}_i \times \mathbf{k}_i$ or :

$$\mathbf{u}_i = \cos(\sigma_i)\,\mathbf{i}_i + \sin(\sigma_i)\,\mathbf{j}_i \tag{2.4}$$

where the symbol $\times$ denotes the vector product between two vectors. By definition, one constraint imposed by the pulley on the cable direction, is defined by the equation:

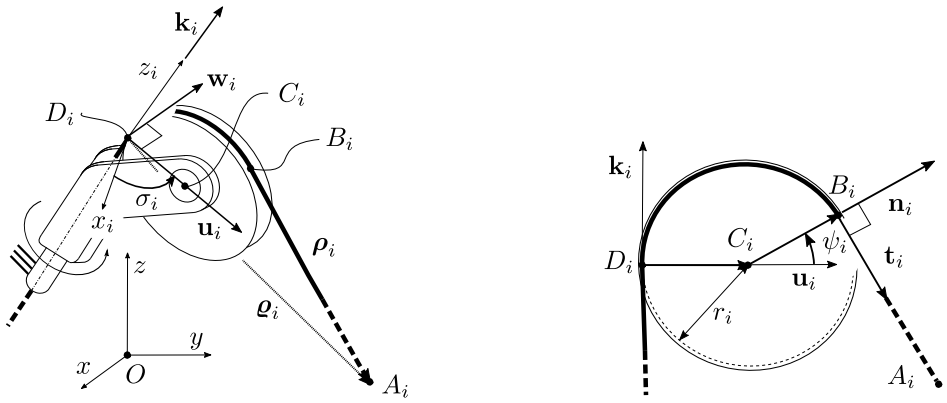
$$\mathbf{w}_i \cdot \boldsymbol{\varrho}_i = 0 \tag{2.5}$$

Accordingly, if $\boldsymbol{\zeta}$ is known, σ_i can be computed from Eq. (2.5) (see App. A.1) as:

$$\sigma_i = \operatorname{atan2}\left(\mathbf{j}_i \cdot \boldsymbol{\varrho}_i, \mathbf{i}_i \cdot \boldsymbol{\varrho}_i\right) \tag{2.6}$$

If $\psi_i \in [-\pi, \pi]$, the *tangency angle*, is the angle between $\mathbf{u}_i$ and the unit vector $\mathbf{n}_i$ in the direction of $B_i - C_i$, we have:

$$\mathbf{n}_i = \cos\left(\psi_i\right)\mathbf{u}_i + \sin\left(\psi_i\right)\mathbf{k}_i \tag{2.7}$$



(a) Swivel pulley local view (b) Pulley plane

Figure 2.2: Swivel Pulley Geometric Model

Additionally, the unit vector $\mathbf{t}_i$ directed as $A_i - B_i$ can be found as $\mathbf{t}_i = \mathbf{w}_i \times \mathbf{n}_i$ or :

$$\mathbf{t}_i = \sin(\psi_i)\,\mathbf{u}_i - \cos(\psi_i)\,\mathbf{k}_i \tag{2.8}$$

If we define the cable vector $\boldsymbol{\rho}_i = \mathbf{b}_i - \mathbf{a}_i$[2], where $\mathbf{b}_i$ is the position vector of B_i in $Oxyz$, the second constraint imposed by the pulley on the cable direction is defined by the equation:

$$\mathbf{n}_i \cdot \boldsymbol{\rho}_i = 0 \tag{2.9}$$

Accordingly, if $\boldsymbol{\zeta}$ is known, ψ_i can be computed from Eq. (2.9) (see App. A.2) as:

$$\psi_i = 2\,\mathrm{atan}\left[\frac{\varrho_{k_i}}{\varrho_{u_i}} + \sqrt{\left(\frac{\varrho_{k_i}}{\varrho_{u_i}}\right)^2 + 1 - \frac{2r_i}{\varrho_{u_i}}}\right] \tag{2.10}$$

where $\varrho_{k_i} = \mathbf{k}_i \cdot \boldsymbol{\varrho}_i$ and $\varrho_{u_i} = \mathbf{u}_i \cdot \boldsymbol{\varrho}_i$

If $l_i > 0$ is the total cable length, comprising the rectilinear part $\|\boldsymbol{\rho}_i\|$ and the arc $\widehat{D_iB_i} = r_i(\pi - \psi)$ wrapped onto the pulley, the i-th cable vector can thus be defined as:

$$\boldsymbol{\rho}_i = \mathbf{a}_i - \mathbf{b}_i = \boldsymbol{\varrho}_i - r_i\,(\mathbf{u}_i + \mathbf{n}_i) \tag{2.11}$$

or:

$$\boldsymbol{\rho}_i = \left[l_i - r_i(\pi - \psi)\right]\mathbf{t_i} = \|\boldsymbol{\rho}_i\|\mathbf{t}_i \tag{2.12}$$

where:

$$\|\boldsymbol{\rho}_i\| = l_i - r_i(\pi - \psi) \tag{2.13}$$

The geometrical constraint imposed by each cable onto the platform is then:

$$\boldsymbol{\rho}_i \cdot \boldsymbol{\rho}_i - \|\boldsymbol{\rho}_i\|^2 = \boldsymbol{\rho}_i \cdot \boldsymbol{\rho}_i - \left[l_i - r_i(\pi - \psi)\right]^2 = 0 \tag{2.14}$$

By simply rearranging and extracting the square root of $\left[l_i - r_i(\pi - \psi)\right]^2$, l_i can be computed from Eq. (2.14), in case $\boldsymbol{\zeta}$ is known, as:

$$l_i = r_i(\pi - \psi_i) + \sqrt{\boldsymbol{\rho}_i \cdot \boldsymbol{\rho}_i} \tag{2.15}$$

[2]Please note that the use of $\boldsymbol{\rho}$ and $\boldsymbol{\varrho}$ in defining position vectors $\mathbf{b}_i - \mathbf{a}_i$ and $\mathbf{d}_i - \mathbf{a}_i$ is intentional, as they are equivalent in case $r_i = 0$, and thus $\mathbf{b}_i = \mathbf{d}_i$.

It should be noted that, in case the pulley radius is negligible or cables are routed into the workspace through eyelets instead of pulleys, namely $r_i = 0$, the model outlined in this Section is still valid and well defined. Obviously, $\boldsymbol{\varrho}_i = \boldsymbol{\rho}_i$.

Notice that geometric equations are not sufficient for the characterization and control of a generic *UACDPR*:

- on the one hand, when cable lengths are assigned, the forward geometric problem, defined by Eq. (2.15), with $i = 1, \ldots, n$, is still underdetermined, since $n < 6$ equations are not sufficient to determine the 6 elements of $\boldsymbol{\zeta}$;
- on the other hand, if the *EE* configuration $\boldsymbol{\zeta}$ is known, the inverse geometric problem, also defined by Eq. (2.15), with $i = 1, \ldots, n$, is completely determined, and the cable lengths corresponding to this specific configuration may be evaluated; however, the said configuration may not be achieved in practice, since infinitely many configurations correspond to the computed cable lengths (and the robot will only reach those that are are compatible with mechanical equilibrium, see Chapter 3 for an example of static configurations computation).

Section 2.2 aims at describing (i) the effects of cable length changes on the *EE* pose (the *EE controlled motion*) and (ii) how the *EE* pose may vary regardless of a cable length variation (the *EE free motion*). The causes of free motion are then described in Sec. 2.3.

2.2 Kinematic Modelling

If $\boldsymbol{\omega}$ is the angular velocity of the *EE*, its twist is $\mathbf{v} = [\dot{\mathbf{p}}^T \, \boldsymbol{\omega}^T]^T$. The angular velocity depends in a non-linear way on the orientation-parameter array $\boldsymbol{\epsilon}$ and linearly from its time derivative $\dot{\boldsymbol{\epsilon}}$, namely:

$$\boldsymbol{\omega} = \mathbf{H}(\boldsymbol{\epsilon})\dot{\boldsymbol{\epsilon}} \tag{2.16}$$

thus the twist relationship with the pose $\boldsymbol{\zeta}$ and its time derivative $\dot{\boldsymbol{\zeta}}$ is given by:

$$\mathbf{v} = \mathbf{D}(\boldsymbol{\epsilon})\dot{\boldsymbol{\zeta}}, \qquad \mathbf{D}(\boldsymbol{\epsilon}) = \begin{bmatrix} \mathbf{I}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{H}(\boldsymbol{\epsilon}) \end{bmatrix} \tag{2.17}$$

with $\mathbf{I}_{3\times3}$ and $\mathbf{0}_{3\times3}$ being identity and null matrices of dimension 3×3. Matrix $\mathbf{H}(\boldsymbol{\epsilon})$ depends on the orientation parametrization that is used, and for tilt-and-torsion angles is given by:

$$\mathbf{H}(\boldsymbol{\epsilon}) = \begin{bmatrix} -c_\phi s_\theta & -s_\phi & c_\phi s_\theta \\ -s_\phi s_\theta & c_\phi & s_\phi s_\theta \\ 1 - c_\theta & 0 & c_\theta \end{bmatrix} \tag{2.18}$$

where $c_\phi = \cos(\phi)$, $s_\phi = \sin(\phi)$, $c_\theta = \cos(\theta)$ and $s_\theta = \sin(\theta)$.

The rate of change of the swivel angle $\dot{\sigma}_i$ can be computed from the time derivative of Eq. (2.5) (see App. A.3) as:

$$\dot{\sigma}_i = \boldsymbol{\xi}_{\sigma_i} \cdot \mathbf{v}, \qquad \boldsymbol{\xi}_{\sigma_i} = \frac{1}{\varrho_{u_i}} \begin{bmatrix} \mathbf{w}_i \\ \mathbf{a}'_i \times \mathbf{w}_i \end{bmatrix} \tag{2.19}$$

whereas the rate of change of the tangency angle $\dot{\psi}_i$ can be computed from the time derivative of Eq. (2.9) (see App. A.4) as:

$$\dot{\psi}_i = \boldsymbol{\xi}_{\psi_i} \cdot \mathbf{v}, \qquad \boldsymbol{\xi}_{\psi_i} = \frac{1}{\|\boldsymbol{\rho}_i\|} \begin{bmatrix} \mathbf{n}_i \\ \mathbf{a}'_i \times \mathbf{n}_i \end{bmatrix} \tag{2.20}$$

Moreover, the rate of change of the cable length $\dot{l}_i$ is computed from the time derivative of Eq. (2.14) (see App. A.5) as:

$$\dot{l}_i = \boldsymbol{\xi}_{l_i} \cdot \mathbf{v}, \qquad \boldsymbol{\xi}_{l_i} = \begin{bmatrix} \mathbf{t}_i \\ \mathbf{a}'_i \times \mathbf{t}_i \end{bmatrix} \tag{2.21}$$

Equations (2.19) to (2.21), for $i = 1, \ldots, n$, can be written in matrix form as:

$$\dot{\boldsymbol{\sigma}} = \boldsymbol{\Xi}_\sigma^T \mathbf{v}, \qquad \boldsymbol{\Xi}_\sigma = \begin{bmatrix} \boldsymbol{\xi}_{\sigma_1} & \ldots & \boldsymbol{\xi}_{\sigma_n} \end{bmatrix} \tag{2.22}$$

$$\dot{\boldsymbol{\psi}} = \boldsymbol{\Xi}_\psi^T \mathbf{v}, \qquad \boldsymbol{\Xi}_\psi = \begin{bmatrix} \boldsymbol{\xi}_{\psi_1} & \ldots & \boldsymbol{\xi}_{\psi_n} \end{bmatrix} \tag{2.23}$$

$$\dot{\mathbf{l}} = \boldsymbol{\Xi}_l^T \mathbf{v}, \qquad \boldsymbol{\Xi}_l = \begin{bmatrix} \boldsymbol{\xi}_{l_1} & \ldots & \boldsymbol{\xi}_{l_n} \end{bmatrix} \tag{2.24}$$

where $\dot{\boldsymbol{\sigma}} = [\dot{\sigma}_1 \ldots, \dot{\sigma}_n]^T$, $\dot{\boldsymbol{\psi}} = [\dot{\psi}_1 \ldots, \dot{\psi}_n]^T$ and $\dot{\mathbf{l}} = [\dot{l}_1 \ldots, \dot{l}_n]^T$. Matrices $\boldsymbol{\Xi}_\sigma$, $\boldsymbol{\Xi}_\psi$ and $\boldsymbol{\Xi}_l$ are kinematic Jacobians, rather than proper Jacobians, since they do not correlate integrable vectors ($\mathbf{v}$ is not integrable in general). Equations (2.22) to (2.24) can be rewritten by means of proper Jacobians if we substitute therein the definition of twist given in Eq. (2.17), namely:

$$\dot{\boldsymbol{\sigma}} = \mathbf{J}_\sigma^T \dot{\boldsymbol{\zeta}}, \qquad \mathbf{J}_\sigma = \mathbf{D}^T \boldsymbol{\Xi}_\sigma \tag{2.25}$$

$$\dot{\boldsymbol{\psi}} = \mathbf{J}_\psi^T \dot{\boldsymbol{\zeta}}, \qquad \mathbf{J}_\psi = \mathbf{D}^T \boldsymbol{\Xi}_\psi \tag{2.26}$$

$$\dot{\mathbf{l}} = \mathbf{J}_l^T \dot{\boldsymbol{\zeta}}, \qquad \mathbf{J}_l = \mathbf{D}^T \boldsymbol{\Xi}_l \tag{2.27}$$

If $\boldsymbol{\alpha}$ is the angular aceleration of the end effector, the twist time derivative is $\dot{\mathbf{v}} = [\ddot{\mathbf{p}}^T \; \boldsymbol{\alpha}^T]^T$. The angular acceleration non-linearly depends on the value of the orientation parameters $\boldsymbol{\epsilon}$, it is bi-linear in $\dot{\boldsymbol{\epsilon}}$, and linear in $\ddot{\boldsymbol{\epsilon}}$:

$$\boldsymbol{\alpha} = \mathbf{H}(\boldsymbol{\epsilon})\ddot{\boldsymbol{\epsilon}} + \dot{\mathbf{H}}(\boldsymbol{\epsilon}, \dot{\boldsymbol{\epsilon}})\dot{\boldsymbol{\epsilon}}, \qquad \dot{\mathbf{H}}(\boldsymbol{\epsilon}, \dot{\boldsymbol{\epsilon}}) = \sum_{i=1}^{3} \frac{\partial \mathbf{H}(\boldsymbol{\epsilon})}{\partial \epsilon_i} \dot{\epsilon}_i \tag{2.28}$$

$$\frac{\partial \mathbf{H}(\boldsymbol{\epsilon})}{\partial \epsilon_1} = \begin{bmatrix} s_\phi s_\theta & -c_\phi & -s_\phi s_\theta \\ -c_\phi s_\theta & -s_\phi & c_\phi s_\theta \\ 0 & 0 & 0 \end{bmatrix}, \qquad \frac{\partial \mathbf{H}(\boldsymbol{\epsilon})}{\partial \epsilon_2} = \begin{bmatrix} -c_\phi c_\theta & 0 & c_\phi c_\theta \\ -s_\phi c_\theta & 0 & s_\phi c_\theta \\ s_\theta & 0 & -s_\theta \end{bmatrix} \tag{2.29}$$

and $\partial \mathbf{H}(\boldsymbol{\epsilon})/\partial \epsilon_3 = \mathbf{0}_{3\times 3}$. Thus, the twist derivative relationship with the pose $\boldsymbol{\zeta}$ and its time derivatives $\dot{\boldsymbol{\zeta}}$ and $\ddot{\boldsymbol{\zeta}}$ is given by:

$$\dot{\mathbf{v}} = \mathbf{D}(\boldsymbol{\epsilon})\ddot{\boldsymbol{\zeta}} + \dot{\mathbf{D}}(\boldsymbol{\epsilon}, \dot{\boldsymbol{\epsilon}})\dot{\boldsymbol{\zeta}}, \qquad \dot{\mathbf{D}}(\boldsymbol{\epsilon}, \dot{\boldsymbol{\epsilon}}) = \begin{bmatrix} \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 3} \\ \mathbf{0}_{3\times 3} & \dot{\mathbf{H}}(\boldsymbol{\epsilon}, \dot{\boldsymbol{\epsilon}}) \end{bmatrix} \tag{2.30}$$

The rate of change $\ddot{\sigma}_i$ of the swivel-angle time-derivative can be computed from the time derivative of Eq. (2.19) as:

$$\ddot{\sigma}_i = \boldsymbol{\xi}_{\sigma_i} \cdot \dot{\mathbf{v}} + \dot{\boldsymbol{\xi}}_{\sigma_i} \cdot \mathbf{v} \tag{2.31}$$

or alternatively (see App. A.6) as:

$$\ddot{\sigma}_i = \mathbf{v}^T \boldsymbol{\xi}'_{\sigma_i} \mathbf{v} + \boldsymbol{\xi}_{\sigma_i}^T \dot{\mathbf{v}} \tag{2.32}$$

$$\boldsymbol{\xi}'_{\sigma_i} = \frac{1}{\varrho_{u_i}} \left(-2\boldsymbol{\xi}_{u_i} \boldsymbol{\xi}_{\sigma_i}^T + \mathbf{A}_{w_i} \right), \qquad \boldsymbol{\xi}_{u_i} = \begin{bmatrix} \mathbf{u}_i \\ \mathbf{a}'_i \times \mathbf{u}_i \end{bmatrix}, \qquad \mathbf{A}_{w_i} = \begin{bmatrix} \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 3} \\ \mathbf{0}_{3\times 3} & \tilde{\mathbf{a}}'_i \tilde{\mathbf{w}}_i \end{bmatrix} \tag{2.33}$$

where $\tilde{(\cdot)}$ denotes the skew-symmetric representation of the vector product.

The rate of change $\ddot{\psi}_i$ of the tangency-angle time-derivative can be computed from the time derivative of Eq. (2.20) as:

$$\ddot{\psi}_i = \dot{\boldsymbol{\xi}}_{\psi_i} \cdot \mathbf{v} + \boldsymbol{\xi}_{\psi_i} \cdot \dot{\mathbf{v}} \tag{2.34}$$

or alternatively (see App. A.7) as:

$$\ddot{\psi}_i = \mathbf{v}^T \boldsymbol{\xi}'_{\psi_i} \mathbf{v} + \boldsymbol{\xi}^T_{\psi_i} \dot{\mathbf{v}} \tag{2.35}$$

$$\boldsymbol{\xi}'_{\psi_i} = \frac{1}{\|\boldsymbol{\rho}_i\|} \left(\varrho_{u_i} \cos(\psi_i) \boldsymbol{\xi}_{\sigma_i} \boldsymbol{\xi}^T_{\sigma_i} - r_i \boldsymbol{\xi}_{\psi_i} \boldsymbol{\xi}^T_{\psi_i} - 2\boldsymbol{\xi}_{l_i} \boldsymbol{\xi}^T_{\psi_i} + \mathbf{A}_{n_i} \right), \quad \mathbf{A}_{n_i} = \begin{bmatrix} \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \tilde{\mathbf{a}}'_i \tilde{\mathbf{n}}_i \end{bmatrix} \tag{2.36}$$

Finally, the rate of change $\ddot{l}_i$ of the cable-length time-derivative can be computed from the time-derivative of Eq. (2.21) as:

$$\ddot{l}_i = \dot{\boldsymbol{\xi}}_{l_i} \cdot \mathbf{v} + \boldsymbol{\xi}_{l_i} \cdot \dot{\mathbf{v}} \tag{2.37}$$

or alternatively (see App. A.8) as:

$$\ddot{l}_i = \mathbf{v}^T \boldsymbol{\xi}'_{l_i} \mathbf{v} + \boldsymbol{\xi}^T_{l_i} \dot{\mathbf{v}} \tag{2.38}$$

$$\boldsymbol{\xi}'_{l_i} = \sin(\psi_i) \varrho_{u_i} \boldsymbol{\xi}_{\sigma_i} \boldsymbol{\xi}^T_{\sigma_i} + \|\boldsymbol{\rho}_i\| \boldsymbol{\xi}_{\psi_i} \boldsymbol{\xi}^T_{\psi_i} + \mathbf{A}_{t_i}, \qquad \mathbf{A}_{t_i} = \begin{bmatrix} \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \tilde{\mathbf{a}}'_i \tilde{\mathbf{t}}_i \end{bmatrix} \tag{2.39}$$

2.2.1 *EE* free and controlled kinematics

Even if all cables are taut, that is, all kinematic constraints are active, and $\boldsymbol{\Xi}_l$ in Eq. (2.24) has full column rank, the *EE* of the *UACDPR* is underactuated, namely the number of its *DoFs* is strictly greater than the number n of *CDPR* actuators. Thus, only n coordinates of the *EE* pose can be controlled by varying the *UACDPR* cable lengths, while the remaining $\lambda = 6 - n$ are to be determined according to the mechanical equilibrium of the platform. In addition, even if the actuators are locked and cable lengths are kept constant, λ freedoms remains. The n controlled coordinates will be referred to as *controlled pose-coordinates* and denoted as $\boldsymbol{\zeta}_c \in \mathbb{R}^n$, whereas the non-controllable coordinates will be referred to as *free pose-coordinates* and denoted as $\boldsymbol{\zeta}_f \in \mathbb{R}^\lambda$. For clarity sake, a 6×6 permutation matrix[3] $\mathbf{P}$ is introduced, so that the array of permutated *EE* generalized coordinates, $\boldsymbol{\zeta}_P$, and its time-derivatives are defined as:

$$\boldsymbol{\zeta}_P = \begin{bmatrix} \boldsymbol{\zeta}_c \\ \boldsymbol{\zeta}_f \end{bmatrix} = \mathbf{P}\boldsymbol{\zeta}, \qquad \dot{\boldsymbol{\zeta}}_P = \mathbf{P}\dot{\boldsymbol{\zeta}}, \qquad \ddot{\boldsymbol{\zeta}}_P = \mathbf{P}\ddot{\boldsymbol{\zeta}} \tag{2.40}$$

The free pose-coordinates are independent configuration variables, since in general are not controllable (i.e. assignable) in an under-actuaded system. Their evolution is determined by the mechanical equilibrium of the *UACDPR* [14], as it will be highlighted in Sec. 2.3. On the other end, the controlled pose-coordinates are assignable and are determined as a function of the system controlled variables, namely the cable lengths $\mathbf{l}$, and the free pose-coordinates, $\boldsymbol{\zeta}_f$:

$$\boldsymbol{\zeta}_c = \boldsymbol{\zeta}_c(\boldsymbol{\zeta}_f, \mathbf{l}) \tag{2.41}$$

[3] A permutation matrix is an orthogonal matrix, that has exactly one entry of 1 in each row and each column, and has 0's elsewhere [49]

Equation (2.41) formally represents the direct geometric problem arising from Eq. (2.15), for $i = 1, \dots, n$. This problem is under-determined, since the value of $\boldsymbol{\zeta}_f$ is not assignable.

The aforementioned coordinate partition is particularly useful when dealing with the *trajectory planning* for an *UACDPR*, but it attains additional kinematic meaning. In fact, the *EE* twist $\mathbf{v}$ can be decomposed into two contributions, namely a *free twist* $\mathbf{v}_f$ and a *controlled twist* $\mathbf{v}_c$, so that:

$$\mathbf{v} = \mathbf{v}_f + \mathbf{v}_c \tag{2.42}$$

The *free twist* is defined as the *EE* twist when the platform is in free motion, and it can be derived as the solution of Eq. (2.24) when $\dot{\mathbf{l}} = \mathbf{0}_{n\times 1}$, namely:

$$\boldsymbol{\Xi}_l^T \mathbf{v}_f = \mathbf{0}_{n\times 1} \tag{2.43}$$

The solution to Eq. (2.43) is readily obtained by considering the right nullspace $\boldsymbol{\Xi}_l^{\perp}$ of matrix $\boldsymbol{\Xi}_l^T$. By definition, the right nullspace of a $(n \times 6)$ matrix is a $(6 \times \lambda)$ matrix such that $\boldsymbol{\Xi}_l^T \boldsymbol{\Xi}_l^{\perp} = \mathbf{0}_{n\times\lambda}$, thus its columns define a basis for the free twist $\mathbf{v}_f$:

$$\mathbf{v}_f = \boldsymbol{\Xi}_l^{\perp} \mathbf{c} \qquad \text{for some } \mathbf{c} \in \mathbb{R}^{\lambda} \tag{2.44}$$

If $\mathbf{J}_l^{\perp}$ is the right nullspace of matrix $\mathbf{J}_l^T$, we also have:

$$\mathbf{v}_f = \mathbf{D}\dot{\boldsymbol{\zeta}} = \mathbf{D}\mathbf{J}_l^{\perp}\mathbf{c}' \qquad \text{for some } \mathbf{c}' \in \mathbb{R}^{\lambda} \tag{2.45}$$

By comparing Eqs. (2.44) and (2.45) and by choosing $\mathbf{c} = \mathbf{c}'$, we have:

$$\boldsymbol{\Xi}_l^{\perp} = \mathbf{D}\mathbf{J}_l^{\perp} \tag{2.46}$$

The coefficients $\mathbf{c}$ coincide with the free-pose coordinates derivative $\dot{\boldsymbol{\zeta}}_f$, if $\boldsymbol{\Xi}_l^{\perp}$ (and thus $\mathbf{J}_l^{\perp}$, cf. Eq.(2.46)) is computed according to the following procedure. First, we define the permuted Jacobian matrix $\mathbf{J}_P$ as:

$$\mathbf{J}_P = \mathbf{P}\mathbf{J}_l = \mathbf{P}\mathbf{D}\boldsymbol{\Xi}_l = \begin{bmatrix} \mathbf{D}_c \\ \mathbf{D}_f \end{bmatrix} \boldsymbol{\Xi}_l = \begin{bmatrix} \mathbf{J}_c \\ \mathbf{J}_f \end{bmatrix} \tag{2.47}$$

where $\mathbf{D}_c \in \mathbb{R}^{n\times 6}$, $\mathbf{D}_f \in \mathbb{R}^{\lambda\times 6}$, $\mathbf{J}_c = \mathbf{D}_c\boldsymbol{\Xi}_l \in \mathbb{R}^{n\times n}$, and $\mathbf{J}_f = \mathbf{D}_f\boldsymbol{\Xi}_l \in \mathbb{R}^{\lambda\times n}$. Then, the right nullspace of the permuted Jacobian matrix $\mathbf{J}_P^{\perp}$ is, by definition:

$$\mathbf{J}_l^T\mathbf{J}_l^{\perp} = \mathbf{J}_l^T\mathbf{P}^T\mathbf{P}\mathbf{J}_l^{\perp} = \mathbf{J}_P^T\mathbf{J}_P^{\perp} = \mathbf{0}_{n\times\lambda}, \qquad \mathbf{J}_P^{\perp} = \mathbf{P}\mathbf{J}_l^{\perp} \tag{2.48}$$

Matrix $\mathbf{J}_P^{\perp}$ can be simbolically computed under the assumption that $\mathbf{J}_c$ is full rank (namely, $\text{rank}(\mathbf{J}_c) = n$) and, thus, invertible. Since most orientation parametrizations of SE(3), among which the tilt and torsion used in this thesis, allow $\text{rank}(D) \geq 5$ and, for an *UACDPR*, $5 \geq n$, we can conclude that $\text{rank}(D) \geq n$. Thus, since $\text{rank}(\boldsymbol{\Xi}_l) = n$, if the pose coordinates are partitioned (namely, $\mathbf{P}$ is chosen) so that $\text{rank}(\mathbf{D}_c) = n$, one may always have $\text{rank}(\mathbf{J}_c) = n$. The expression of matrix $\mathbf{J}_P^{\perp}$ can be derived from Eq. (2.27) with $\mathbf{l} = \mathbf{0}_{n\times 1}$:

$$\mathbf{J}_l^T\dot{\boldsymbol{\zeta}} = \mathbf{J}_P^T\dot{\boldsymbol{\zeta}}_P = \mathbf{J}_c^T\dot{\boldsymbol{\zeta}}_c + \mathbf{J}_f^T\dot{\boldsymbol{\zeta}}_f = \mathbf{0}_{n\times 1} \tag{2.49}$$

where definitions in Eqs. (2.40) and (2.48) have been used. Then:

$$\dot{\boldsymbol{\zeta}}_c = -\mathbf{J}_c^{-T}\mathbf{J}_f^T\dot{\boldsymbol{\zeta}}_f \tag{2.50}$$

and finally:

$$\dot{\boldsymbol{\zeta}}_P = \begin{bmatrix} \dot{\boldsymbol{\zeta}}_c \\ \dot{\boldsymbol{\zeta}}_f \end{bmatrix} = \begin{bmatrix} -\mathbf{J}_c^{-T}\mathbf{J}_f^T \\ \mathbf{I}_{\lambda\times\lambda} \end{bmatrix} \dot{\boldsymbol{\zeta}}_f = \mathbf{J}_P^{\perp}\dot{\boldsymbol{\zeta}}_f, \qquad \mathbf{J}_P^{\perp} = \begin{bmatrix} -\mathbf{J}_c^{-T}\mathbf{J}_f^T \\ \mathbf{I}_{\lambda\times\lambda} \end{bmatrix} \tag{2.51}$$

It should be noted that the expression of $\dot{\boldsymbol{\zeta}}_P$ provided in Eq. (2.51) is valid *only* in free motion. In the end, one has:

$$\mathbf{v}_f = \mathbf{D}\dot{\boldsymbol{\zeta}} = \mathbf{D}\mathbf{P}^T\dot{\boldsymbol{\zeta}}_P = \mathbf{D}\mathbf{P}^T\mathbf{J}_P^{\perp}\dot{\boldsymbol{\zeta}}_f = \mathbf{D}\mathbf{J}_l^{\perp}\dot{\boldsymbol{\zeta}}_f = \boldsymbol{\Xi}_l^{\perp}\dot{\boldsymbol{\zeta}}_f, \qquad \boldsymbol{\Xi}_l^{\perp} = \mathbf{D}\mathbf{P}^T\mathbf{J}_P^{\perp} = \mathbf{D}\mathbf{J}_l^{\perp} \tag{2.52}$$

where $\mathbf{c} = \dot{\boldsymbol{\zeta}}_f$. The *controlled twist* is defined as the *EE* twist due to cable actuation only, that is, the one resulting from a zero free-pose coordinates derivative, $\dot{\boldsymbol{\zeta}}_f = \mathbf{0}_{\lambda\times 1}$. The expression of $\dot{\boldsymbol{\zeta}}_P$ in this case can be straightforwardly derived as the solution of Eq. (2.27) by setting $\dot{\boldsymbol{\zeta}}_f = \mathbf{0}_{\lambda\times 1}$ and recalling position (2.47), namely:

$$\dot{\mathbf{l}} = \mathbf{J}_l^T\dot{\boldsymbol{\zeta}} = \mathbf{J}_P^T\dot{\boldsymbol{\zeta}}_P = \mathbf{J}_c^T\dot{\boldsymbol{\zeta}}_c \qquad \Longrightarrow \qquad \dot{\boldsymbol{\zeta}}_c = \mathbf{J}_c^{-T}\dot{\mathbf{l}} \tag{2.53}$$

and consequently:

$$\dot{\boldsymbol{\zeta}}_P = \begin{bmatrix} \mathbf{J}_c^{-T} \\ \mathbf{0}_{\lambda\times n} \end{bmatrix} \dot{\mathbf{l}} = \mathbf{J}_P^{\parallel}\dot{\mathbf{l}}, \qquad \mathbf{J}_P^{\parallel} = \begin{bmatrix} \mathbf{J}_c^{-T} \\ \mathbf{0}_{\lambda\times n} \end{bmatrix} \tag{2.54}$$

Analogously to Eq. (2.51), the expression of $\dot{\boldsymbol{\zeta}}_P$ provided in Eq. (2.54) is valid *only* for $\dot{\boldsymbol{\zeta}}_f = \mathbf{0}_{\lambda\times 1}$. Finally, one has:

$$\mathbf{v}_c = \mathbf{D}\dot{\boldsymbol{\zeta}} = \mathbf{D}\mathbf{P}^T\dot{\boldsymbol{\zeta}}_P = \mathbf{D}\mathbf{P}^T\mathbf{J}_P^{\parallel}\dot{\mathbf{l}} = \mathbf{D}\mathbf{J}_l^{\parallel}\dot{\mathbf{l}} = \boldsymbol{\Xi}_l^{\parallel}\dot{\mathbf{l}}, \qquad \boldsymbol{\Xi}_l^{\parallel} = \mathbf{D}\mathbf{P}^T\mathbf{J}_P^{\parallel}, \qquad \mathbf{J}_l^{\parallel} = \mathbf{P}^T\mathbf{J}_l^{\parallel} \tag{2.55}$$

The results of Eqs. (2.52) and (2.55) can be substituted back into Eq. (2.42) in order to highlight the *EE* twist linear dependence on the robot controllable-coordinate derivative, $\dot{\mathbf{l}}$, and the free pose-coordinate derivatives, $\dot{\boldsymbol{\zeta}}_f$:

$$\mathbf{v} = \mathbf{v}_f + \mathbf{v}_c = \boldsymbol{\Xi}_l^{\perp}\dot{\boldsymbol{\zeta}}_f + \boldsymbol{\Xi}_l^{\parallel}\dot{\mathbf{l}} \tag{2.56}$$

It can be straightforwardly verified that the expression of $\mathbf{v}$ in Eq. (2.56) verifies *EE* first-order kinematics, expressed in Eq. (2.24):

$$\dot{\mathbf{l}} = \boldsymbol{\Xi}_l^T\mathbf{v} = \boldsymbol{\Xi}_l^T\left(\boldsymbol{\Xi}_l^{\perp}\dot{\boldsymbol{\zeta}}_f + \boldsymbol{\Xi}_l^{\parallel}\dot{\mathbf{l}}\right) = \underbrace{\boldsymbol{\Xi}_l^T\boldsymbol{\Xi}_l^{\perp}}_{\mathbf{0}_{n\times\lambda}}\dot{\boldsymbol{\zeta}}_f + \underbrace{\boldsymbol{\Xi}_l^T\boldsymbol{\Xi}_l^{\parallel}}_{\mathbf{I}_{n\times n}}\dot{\mathbf{l}} = \dot{\mathbf{l}} \tag{2.57}$$

Of course, since $\mathbf{v} = \mathbf{D}\dot{\boldsymbol{\zeta}}$, one also has:

$$\dot{\boldsymbol{\zeta}} = \mathbf{J}_l^{\perp}\dot{\boldsymbol{\zeta}}_f + \mathbf{J}_l^{\parallel}\dot{\mathbf{l}} \tag{2.58}$$

Although it is *always* possible to find a permutation matrix $\mathbf{P}$ which allows for the sound definition of free and controllable twists in one particular *EE* configuration, the same permutation matrix may not provide the expected result throughout the entire manipulator workspace. In fact, for a given *EE* pose parametrization, some parameters may be locally controllable or not, depending on the rank of $\mathbf{J}_c$. In case the latter rank is less than n, but the *EE* still has n controllable *DoFs* and λ free *DoFs*, a parametrization singularity is encountered. In this case, the choice of the free pose-coordinates must be changed so that $\mathbf{J}_c$ preserves full rank. Should $\boldsymbol{\Xi}_l$ rank become less than n, instead, a true kinematic singularity is encountered, and the number of controllable *DoFs* is $\text{rank}(\boldsymbol{\Xi}_l) = n' < n$. It should be noted that, upon (locally) re-defining controllable and

free coordinates, so that $\boldsymbol{\zeta}_c \in \mathbb{R}^{n'}$ and $\boldsymbol{\zeta}_f \in \mathbb{R}^{\lambda'}$, where $\lambda' = 6 - n'$, the model outlined in this Section is still sound.

An analytical expression of the *EE* twist time-derivative is useful in *EE* dynamic modelling, as it will be shown in Sec. 2.3, and it can be derived by differentiating Eq. (2.56) w.r.t. time:

$$\dot{\mathbf{v}} = \boldsymbol{\Xi}_l^{\perp} \ddot{\boldsymbol{\zeta}}_f + \boldsymbol{\Xi}_l^{\parallel} \ddot{\mathbf{l}} + \dot{\boldsymbol{\Xi}}_l^{\perp} \dot{\boldsymbol{\zeta}}_f + \dot{\boldsymbol{\Xi}}_l^{\parallel} \dot{\mathbf{l}} \tag{2.59}$$

where $\dot{\boldsymbol{\Xi}}_l^{\perp}$ and $\dot{\boldsymbol{\Xi}}_l^{\parallel}$ can be derived from their definition in Eqs. (2.52) and (2.55) as:

$$\dot{\boldsymbol{\Xi}}_l^{\perp} = \dot{\mathbf{D}}\mathbf{P}^T \mathbf{J}_P^{\perp} + \mathbf{D}\mathbf{P}^T \dot{\mathbf{J}}_P^{\perp}, \qquad \dot{\boldsymbol{\Xi}}_l^{\parallel} = \dot{\mathbf{D}}\mathbf{P}^T \mathbf{J}_P^{\parallel} + \mathbf{D}\mathbf{P}^T \dot{\mathbf{J}}_P^{\parallel} \tag{2.60}$$

and:

$$\dot{\mathbf{J}}_P^{\perp} = \begin{bmatrix} \mathbf{J}_c^{-T} \left(\dot{\mathbf{J}}_c^T \mathbf{J}_c^{-T} \mathbf{J}_f^T - \dot{\mathbf{J}}_f^T \right) \\ \mathbf{0}_{\lambda \times \lambda} \end{bmatrix}, \qquad \dot{\mathbf{J}}_P^{\parallel} = \begin{bmatrix} -\mathbf{J}_c^{-T} \dot{\mathbf{J}}_c^T \mathbf{J}_c^{-T} \\ \mathbf{0}_{\lambda \times n} \end{bmatrix} \tag{2.61}$$

At last, $\dot{\mathbf{J}}_c$ and $\dot{\mathbf{J}}_f$ can be computed by differentiating Eq. (2.47) w.r.t. time, namely:

$$\dot{\mathbf{J}}_P = \begin{bmatrix} \dot{\mathbf{J}}_c \\ \dot{\mathbf{J}}_f \end{bmatrix} = \mathbf{P} \left(\dot{\mathbf{D}} \boldsymbol{\Xi}_l + \mathbf{D} \dot{\boldsymbol{\Xi}}_l \right), \qquad \dot{\boldsymbol{\Xi}}_l = \begin{bmatrix} \dot{\boldsymbol{\xi}}_{l_1} & \dots & \dot{\boldsymbol{\xi}}_{l_n} \end{bmatrix} \tag{2.62}$$

and $\dot{\boldsymbol{\xi}}_{l_i} = \boldsymbol{\xi}'_{l_i} \mathbf{v}$ from Eqs. (2.37) and (2.38). Equations from (2.59) to (2.62) highlight that the twist time-derivative is linearly dependent on the second-order time derivative of the controlled and free variables of the *CDPR* (namely $\ddot{\mathbf{l}}$ and $\ddot{\boldsymbol{\zeta}}_f$, respectively), and it is bi-linear in their first-order time derivative ($\dot{\mathbf{l}}$ and $\dot{\boldsymbol{\zeta}}_f$, respectively).

2.3 Dynamic Modelling

The non-linear dynamic model of the *UACDPR* emerges from the *EE* mechanical equilibrium, subject to cable constraints, inertial actions, and a generic external wrench:

$$\mathbf{M}\dot{\mathbf{v}} + \mathbf{C}\mathbf{v} = -\boldsymbol{\Xi}_l \boldsymbol{\tau} + \mathbf{f} \tag{2.63}$$

$$\begin{aligned} \mathbf{M} &= \begin{bmatrix} m\mathbf{I}_{3\times 3} & -m\tilde{\mathbf{s}}' \\ m\tilde{\mathbf{s}}' & \mathbf{I}_P \end{bmatrix}, \quad \mathbf{I}_P = \mathbf{I}_G - m\tilde{\mathbf{s}}'\tilde{\mathbf{s}}' \\ \mathbf{C} &= \begin{bmatrix} \mathbf{0}_{3\times 3} & -m\tilde{\boldsymbol{\omega}}\tilde{\mathbf{s}}' \\ \mathbf{0}_{3\times 3} & \tilde{\boldsymbol{\omega}}\mathbf{I}_P \end{bmatrix}, \quad \mathbf{f} = \begin{bmatrix} \boldsymbol{\phi} \\ \tilde{\mathbf{e}}'\boldsymbol{\phi} + \boldsymbol{\mu} \end{bmatrix} \end{aligned} \tag{2.64}$$

where m is the *EE* mass, $\mathbf{I}_G = \mathbf{R}\mathbf{I}'_G\mathbf{R}^T$ is the (variable) *EE* inertia tensor about its center of mass G expressed in the inertial frame, $\mathbf{I}'_G$ is the (constant) *EE* matrix in the mobile frame; $\boldsymbol{\tau} \in \mathbb{R}^n$ is an array containing the cable tension magnitudes, and $\mathbf{f} \in \mathbb{R}^6$ is a generic external wrench, resulting from a force $\boldsymbol{\phi}$ applied in point E and a moment $\boldsymbol{\mu}$ directed along $\boldsymbol{\phi}$ (Fig. 2.3). Vectors $\mathbf{s}'$ and $\mathbf{e}'$ point from P to G and E, respectively, and are computed in the inertial frame as:

$$\mathbf{s}' = \mathbf{R}\,{}^P\mathbf{s}', \qquad \mathbf{e}' = \mathbf{R}\,{}^P\mathbf{e}' \tag{2.65}$$

with ${}^P\mathbf{s}'$ and ${}^P\mathbf{e}'$ being their coordinates in the mobile frame. In general, both the position of G and the position of E w.r.t. to point P are not constant, but may change depending on the task that the *EE* is required to perform.

Since the constraint actions imposed by the cables, namely $-\boldsymbol{\Xi}_l\boldsymbol{\tau}$, are not sufficient for constraining the *EE* motion, the robot *EE* is under-constrained. In other words, the dimension of the constraint space spanned by passive and active constraints acting on the *EE* is smaller than six [13], and the *EE* preserves some degrees of freedom which cannot be actively controlled by cables.

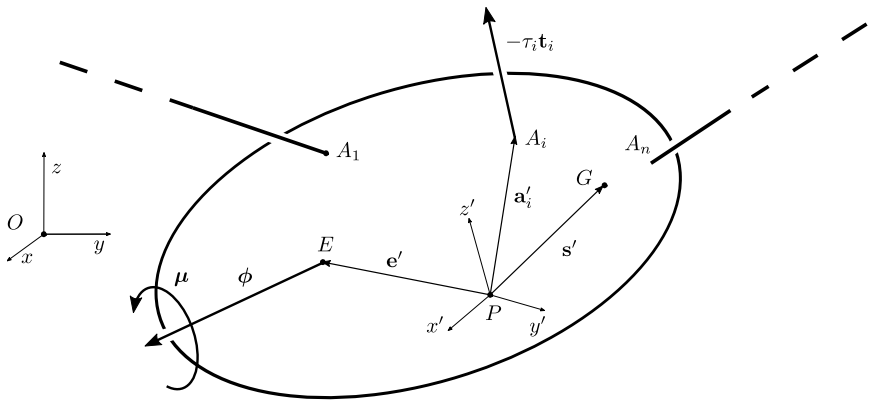


Figure 2.3: *EE* Free-Body Diagram

2.3.1 *EE* internal-dynamics

In order to understand how the *EE* pose-coordinates evolve regardless of the cable constraint actions, Eq. (2.63) can be pre-multiplyed by $\boldsymbol{\Xi}_l^{\perp T}$. Since by definition $\boldsymbol{\Xi}_l^{\perp T}\boldsymbol{\Xi}_l = \mathbf{0}_{\lambda\times n}$, we have:

$$\boldsymbol{\Xi}_l^{\perp T}\mathbf{M}\dot{\mathbf{v}} + \boldsymbol{\Xi}_l^{\perp T}\mathbf{C}\mathbf{v} = \boldsymbol{\Xi}_l^{\perp T}\mathbf{f} \tag{2.66}$$

If we substitute the definitions of $\mathbf{v}$ and $\dot{\mathbf{v}}$ in Eqs. (2.17) and (2.30) in Eq. (2.66), we get:

$$\boldsymbol{\Xi}_l^{\perp T}\mathbf{M}\mathbf{D}\ddot{\boldsymbol{\zeta}} + \boldsymbol{\Xi}_l^{\perp T}\left(\mathbf{M}\dot{\mathbf{D}} + \mathbf{C}\mathbf{D}\right)\dot{\boldsymbol{\zeta}} = \boldsymbol{\Xi}_l^{\perp T}\mathbf{f} \tag{2.67}$$

The relationship between the free coordinates of the end-effector and the controlled ones is highlighted by substituting Eq. (2.40) in Eq. (2.67), yielding:

$$\mathbf{M}_{cP}^{\perp}\ddot{\boldsymbol{\zeta}}_c + \mathbf{M}_{fP}^{\perp}\ddot{\boldsymbol{\zeta}}_f + \mathbf{C}_{cP}^{\perp}\dot{\boldsymbol{\zeta}}_c + \mathbf{C}_{fP}^{\perp}\dot{\boldsymbol{\zeta}}_f + \mathbf{f}^{\perp} = \mathbf{0}_{\lambda\times 1}, \qquad \mathbf{f}^{\perp} = -\boldsymbol{\Xi}_l^{\perp T}\mathbf{f} \tag{2.68}$$

where:

$$\boldsymbol{\Xi}_l^{\perp T}\mathbf{M}\mathbf{D}\mathbf{P}^T = \begin{bmatrix}\mathbf{M}_{cP}^{\perp} & \mathbf{M}_{fP}^{\perp}\end{bmatrix}, \qquad \mathbf{M}_{cP}^{\perp} \in \mathbb{R}^{\lambda\times n}, \mathbf{M}_{fP}^{\perp} \in \mathbb{R}^{\lambda\times\lambda} \tag{2.69}$$

$$\boldsymbol{\Xi}_l^{\perp T}\left(\mathbf{M}\dot{\mathbf{D}} + \mathbf{C}\mathbf{D}\right)\mathbf{P}^T = \begin{bmatrix}\mathbf{C}_{cP}^{\perp} & \mathbf{C}_{fP}^{\perp}\end{bmatrix}, \qquad \mathbf{C}_{cP}^{\perp} \in \mathbb{R}^{\lambda\times n}, \mathbf{C}_{fP}^{\perp} \in \mathbb{R}^{\lambda\times\lambda} \tag{2.70}$$

Equation (2.68) allows one to analyze how the control of some *EE* coordinates influence the free ones, and it is referred to as a second-order non-holonomic constraint, also known as *internal dynamics*, arising from the under-constrained nature of the system [43]. This formulation is fundamental in order to analyze how an assigned trajectory in the task-space, which must be limited to controllable *DoFs*, influence the *EE* pose.

In case a trajectory in the actuator-space is assigned, its influence on the *EE* pose evolution is analyzed by substituting the expressions of $\mathbf{v}$ and $\dot{\mathbf{v}}$ in Eqs. (2.56) and (2.59) in Eq. (2.66):

$$\mathbf{M}_f^{\perp}\ddot{\boldsymbol{\zeta}}_f + \mathbf{M}_l^{\perp}\ddot{\mathbf{l}} + \mathbf{C}_f^{\perp}\dot{\boldsymbol{\zeta}}_f + \mathbf{C}_l^{\perp}\dot{\mathbf{l}} + \mathbf{f}^{\perp} = \mathbf{0}_{\lambda\times 1} \tag{2.71}$$

where:

$$\mathbf{M}_f^{\perp} = \boldsymbol{\Xi}_l^{\perp T}\mathbf{M}\boldsymbol{\Xi}_l^{\perp} \in \mathbb{R}^{\lambda\times\lambda}, \qquad \mathbf{M}_l^{\perp} = \boldsymbol{\Xi}_l^{\perp T}\mathbf{M}\boldsymbol{\Xi}_l^{\parallel} \in \mathbb{R}^{\lambda\times n} \tag{2.72}$$

$$\mathbf{C}_f^{\perp} = \boldsymbol{\Xi}_l^{\perp T}\mathbf{M}\dot{\boldsymbol{\Xi}}_l^{\perp} + \boldsymbol{\Xi}_l^{\perp T}\mathbf{C}\boldsymbol{\Xi}_l^{\perp} \in \mathbb{R}^{\lambda\times\lambda}, \qquad \mathbf{C}_l^{\perp} = \boldsymbol{\Xi}_l^{\perp T}\mathbf{M}\dot{\boldsymbol{\Xi}}_l^{\parallel} + \boldsymbol{\Xi}_l^{\perp T}\mathbf{C}\boldsymbol{\Xi}_l^{\parallel} \in \mathbb{R}^{\lambda\times n} \tag{2.73}$$

Moreover, in case actuators are locked (namely $\dot{\mathbf{l}} = \ddot{\mathbf{l}} = \mathbf{0}_{n\times 1}$), motion equations of the free-pose coordinates, namely the *free-motion internal dynamics*, are obtained from Eq. (2.71):

$$\mathbf{M}_f^{\perp}\ddot{\boldsymbol{\zeta}}_f + \mathbf{C}_f^{\perp}\dot{\boldsymbol{\zeta}}_f + \mathbf{f}^{\perp} = \mathbf{0}_{\lambda\times 1} \tag{2.74}$$

Depending on the specific application, and possibly additional mathematical or physical constraints, Eqs. (2.68), (2.71), or (2.74), can be numerically or analytically solved with suitable tools:

- in Sec. 4.1, Eq. (2.74) is linearized about equilibrium configurations, and it is shown how to determine a closed-form analytical solution that describes manipulator free motion about the said equilibrium configurations;
- in Sec. 5.1, Eq. (2.68) is used in order to compute the free coordinates evolution when the system follows a known trajectory of the controlled coordinates; to this end, Eq. (2.68) is transformed in state-space representation, and the initial value problem, defined by combining the resulting ordinary differential equation with initial conditions (an equilibrium state), is solved by means of state of Runge-Kutta methods;
- in Sec. 5.2, Eq. (2.68) is used in order to compute a trajectory of the controlled coordinates resulting in rest-to-rest motions; to this end, Eq. (2.68) is again transformed in state-space representation, but a boundary value problem, defined by combining the ordinary differential equation with initial and final equilibrium conditions, is solved by means of collocation methods.

2.3.2 Cable tension computation

The knowledge of cable tensions is fundamental in any cable-robotic application, since cables can only sustain tensile forces; for an *UACDPR*, they cannot be assigned as in their over-actuated counterparts, but they can be calculated in closed form from the manipulator dynamics in case the *EE* motion is known. In the following, it will be shown how cable tensions are influenced by a prescribed trajectory: in case controlled pose-coordinates are prescribed, cable tensions can be calculated by pre-multiplying Eq. (2.63) by $-\boldsymbol{\Xi}_l^{\parallel T}$ and substituting $\mathbf{v}$ and $\dot{\mathbf{v}}$ as in Eqs. (2.17) and (2.30), yielding:

$$\mathbf{M}_{cP}^{\parallel}\ddot{\boldsymbol{\zeta}}_c + \mathbf{M}_{fP}^{\parallel}\ddot{\boldsymbol{\zeta}}_f + \mathbf{C}_{cP}^{\parallel}\dot{\boldsymbol{\zeta}}_c + \mathbf{C}_{fP}^{\parallel}\dot{\boldsymbol{\zeta}}_f + \mathbf{f}^{\parallel} = \boldsymbol{\tau}, \qquad \mathbf{f}^{\parallel} = \boldsymbol{\Xi}_l^{\parallel T}\mathbf{f} \tag{2.75}$$

where:

$$-\boldsymbol{\Xi}_l^{\parallel T}\mathbf{M}\mathbf{D}\mathbf{P}^T = \begin{bmatrix}\mathbf{M}_{cP}^{\parallel} & \mathbf{M}_{fP}^{\parallel}\end{bmatrix}, \qquad \mathbf{M}_{cP}^{\parallel} \in \mathbb{R}^{n\times n}, \mathbf{M}_{fP}^{\parallel} \in \mathbb{R}^{n\times\lambda} \tag{2.76}$$

$$-\boldsymbol{\Xi}_l^{\parallel T}\left(\mathbf{M}\dot{\mathbf{D}} + \mathbf{C}\mathbf{D}\right)\mathbf{P}^T = \begin{bmatrix}\mathbf{C}_{cP}^{\parallel} & \mathbf{C}_{fP}^{\parallel}\end{bmatrix}, \qquad \mathbf{C}_{cP}^{\parallel} \in \mathbb{R}^{n\times n}, \mathbf{C}_{fP}^{\parallel} \in \mathbb{R}^{n\times\lambda} \tag{2.77}$$

Alternatively, if we pre-multiply Eq. (2.63) by $-\boldsymbol{\Xi}_l^{\parallel T}$ and we substitute the expressions of $\mathbf{v}$ and $\dot{\mathbf{v}}$ in Eqs. (2.56) and (2.59), the explicit relationship between cable tensions and cable-length variations is obtained:

$$\mathbf{M}_f^{\parallel}\ddot{\boldsymbol{\zeta}}_f + \mathbf{M}_l^{\parallel}\ddot{\mathbf{l}} + \mathbf{C}_f^{\parallel}\dot{\boldsymbol{\zeta}}_f + \mathbf{C}_l^{\parallel}\dot{\mathbf{l}} + \mathbf{f}^{\parallel} = \boldsymbol{\tau} \tag{2.78}$$

where:

$$\mathbf{M}_f^{\parallel} = -\mathbf{M}_l^{\perp T}, \qquad \mathbf{M}_l^{\parallel} = -\boldsymbol{\Xi}_l^{\parallel T}\mathbf{M}\boldsymbol{\Xi}_l^{\parallel} \in \mathbb{R}^{n\times n} \tag{2.79}$$

$$\mathbf{C}_f^{\parallel} = -(\boldsymbol{\Xi}_l^{\parallel T}\mathbf{M}\dot{\boldsymbol{\Xi}}_l^{\perp} + \boldsymbol{\Xi}_l^{\parallel T}\mathbf{C}\boldsymbol{\Xi}_l^{\perp}) \in \mathbb{R}^{n\times\lambda}, \qquad \mathbf{C}_l^{\parallel} = -(\boldsymbol{\Xi}_l^{\parallel T}\mathbf{M}\dot{\boldsymbol{\Xi}}_l^{\parallel} + \boldsymbol{\Xi}_l^{\parallel T}\mathbf{C}\boldsymbol{\Xi}_l^{\parallel}) \in \mathbb{R}^{n\times n} \tag{2.80}$$

In case actuators are locked, cable tensions are computed from the free-pose coordinates as:

$$\mathbf{M}_f^{\parallel}\ddot{\boldsymbol{\zeta}}_f + \mathbf{C}_f^{\parallel}\dot{\boldsymbol{\zeta}}_f + \mathbf{f}^{\parallel} = \boldsymbol{\tau} \tag{2.81}$$

2.4 Static Modelling

While in a completely-actuated *CDPR* the *EE* pose is determined in a purely geometric way by assigning the cable lengths (provided that all cables are under tension), the pose of a *UACDPR* depends on both cable lengths and mechanical equilibrium equations [14, 15]. Only a set of n coordinates can be controlled, while the remaining ones are determined by considering geometric and mechanical equilibrium equations. Even though Statics is a particular case of Dynamics, it deserves special attention since its study allows for the determination of *EE* stable equilibrium configurations.

2.4.1 Geometrico-Static problems

The static model of the platform is simply given by Eq. (2.63) where $\dot{\mathbf{v}} = \mathbf{v} = \mathbf{0}_{6\times1}$, namely:

$$\boldsymbol{\Xi}_l\boldsymbol{\tau} = \mathbf{f} \tag{2.82}$$

In order to understand how the equilibrium pose of the *EE* is influenced by statics, Eq. (2.82) can be pre-multiplied by $\boldsymbol{\Xi}^{\perp T}$ (as in Eq. (2.66)), yielding:

$$\boldsymbol{\Xi}^{\perp T}\mathbf{f} = -\mathbf{f}^{\perp} = \mathbf{0}_{\lambda\times1} \tag{2.83}$$

We will refer to Eq. (2.83) as the *Static Constraint* that *EE* pose variables must satisfy when equilibrium is attained. The value of cable tensions when $\dot{\mathbf{v}} = \mathbf{v} = \mathbf{0}_{6\times1}$ is calculated, as in Eq. (2.75), by pre-multiplying Eq. (2.82) by $\boldsymbol{\Xi}^{\parallel T}$:

$$\boldsymbol{\tau} = \boldsymbol{\Xi}^{\parallel T}\mathbf{f} = \mathbf{f}^{\parallel} \tag{2.84}$$

In case an inverse problem is formulated, only a subset of the pose $\boldsymbol{\zeta}$ is assignable, namely $\boldsymbol{\zeta}_c \in \mathbb{R}^n$, and both $\boldsymbol{\zeta}_f \in \mathbb{R}^\lambda$ and $\mathbf{l} \in \mathbb{R}^n$ are to be determined: geometric equations are n, thus insufficient to determine $\lambda + n = 6$ unknowns. If the problem is direct, the assigned variable is $\mathbf{l} \in \mathbb{R}^n$, and $\boldsymbol{\zeta} \in \mathbb{R}^{n+\lambda}$ has to be found, thus the number of unknown does not change and the geometric problem is still underdetermined. By coupling geometric equations and statics, the so-called *Geometrico-Static* problems of *UACDPRs* [14, 15] are defined, whose aim is to determine equilibrium configurations, namely a set $(\boldsymbol{\zeta}, \mathbf{l}) = (\boldsymbol{\zeta}_0, \mathbf{l}_0)$, such that Eqs. (2.14) for $i = 1, \cdots, n$ and (2.83) are satisfied, and $\boldsymbol{\tau}$ in Eq. (2.84) is stricly positive.

Although the aforementioned definition of the Geometrico-Static problem is general, it may be specialized for the inverse or forward problem in order to simplify its solution procedure. In fact, for the inverse problem (*IGSP* in short), it is simpler to:

- first, determine the value of $\boldsymbol{\zeta}_f$ which satisfies the *Static Constraint* for an assigned $\boldsymbol{\zeta}_c$, by solving the non-linear system of λ equations in Eq. (2.83), namely:

  $$\mathbf{f}^{\perp}(\boldsymbol{\zeta}_f) = \mathbf{0}_{\lambda\times1} \tag{2.85}$$

  The computation of $\boldsymbol{\zeta}_f$ can be carried out by using a numerical algorithm (e.g. Newton Raphson scheme);

- then, evaluate cable tensions from Eq. (2.84): the evaluation of $\mathbf{f}^{\parallel}$ is straightforward after determining $\boldsymbol{\zeta}_f$, since all pose coordinates are known;

- finally, if $\boldsymbol{\tau} \succeq \mathbf{0}_{n\times 1}$ ($\succeq$ is used to indicate element-wise inequality), that is, all constraints are active, evaluate $\mathbf{l}$ according to Eq. (2.15).

The forward problem (*FGSP*) is slightly more intricated to formulate, because it is not possible to decouple geometry and statics in the determination of the *EE* pose. Accordingly, we suggest to:

- first, determine the value of $\boldsymbol{\zeta}$ which satisfies the static equilibrium and the geometry of the *UACDPR*, by solving the non-linear system of 6 equations:

$$\begin{cases} l_1(\boldsymbol{\zeta}) - l_{1,0} = 0 \\ \vdots \\ l_n(\boldsymbol{\zeta}) - l_{n,0} = 0 \\ \mathbf{f}^{\perp}(\boldsymbol{\zeta}) = \mathbf{0}_{\lambda\times 1} \end{cases} \tag{2.86}$$

  where $l_i(\boldsymbol{\zeta})$ is calculated as in Eq. (2.15), and $l_{i,0}$ is the assigned length of the i-th cable;

- then, evaluate cable tensions by Eq. (2.84): if $\boldsymbol{\tau} \succeq \mathbf{0}_{n\times 1}$ the solution is an equilibrium configuration.

Alternative formulations to Eq. (2.86) are possible [33, 34], which are particularly suitable for the application of continuation methods or interval analysis, but it is the author opinion that Eq. (2.86) provides some advantages if iterative non-linear solvers (e.g. Newton-Raphson, Levenberg-Marquardt) are employed for the numerical computation of one of its possibly many solutions [44].

It should be noted that both the inverse and the forward problem may admit multiple real solutions: the inverse geometrico-static problem was investigated thoroughly for 3- [36] and 4-cable [35] *UACDPRs*, whereas the direct geometrico-static problem was studied for 3- [31, 32, 50], and general $n-$cable [33, 34] *UACDPRs*, and thus are not the object of this thesis.

2.4.2 Free-Motion and Controlled-Motion stiffness

The equilibrium configuration of a *UACDPR* may change for two reasons: cable lengths are varied, or the external wrench changes. The external wrench depends both on modeled effects, such as external force fields or task-related actions, and thus depends on the *EE* configuration, and on un-modeled disturbances, such as unexpected interactions with the surrounding environment. For the purpose of this analysis, we will consider that the external wrench is dependent on the *EE* configuration only, namely $\mathbf{f} = \mathbf{f}(\boldsymbol{\zeta})$, and disturbances are quick and temporary phenomena. In other words, disturbances may temporary alter equilibrium, but the system is nominally not affected by them. The aim of this Subsection is to provide modelling tools for the analysis of the variation of equilibrium configurations due to changes in the cable lenghts or the external wrench..

In case equilibrium is altered, its variation is studied by differentiating the static constraint $\mathbf{f}^{\perp} = \mathbf{0}_{\lambda\times 1}$ in Eq. (2.83), namely:

$$\mathrm{d}\mathbf{f}^{\perp} = \frac{\partial \mathbf{f}^{\perp}}{\partial \boldsymbol{\zeta}}\mathrm{d}\boldsymbol{\zeta} = \mathbf{0}_{\lambda\times 1} \tag{2.87}$$

If we consider Eq. (2.58), namely, $\dot{\boldsymbol{\zeta}} = \mathbf{J}_l^{\perp}\dot{\boldsymbol{\zeta}}_f + \mathbf{J}_l^{\parallel}\dot{\mathbf{l}}$, we can deduce that a pose variation $\mathrm{d}\boldsymbol{\zeta}$ may be produced by a variations of either cable lengths or free-pose coordinates, namely:

$$\mathrm{d}\boldsymbol{\zeta} = \mathbf{J}_l^{\perp}\mathrm{d}\boldsymbol{\zeta}_f + \mathbf{J}_l^{\parallel}\mathrm{d}\mathbf{l} \tag{2.88}$$

So, if we substitute Eq. (2.88) in Eq.(2.87), we get:

$$\mathrm{d}\mathbf{f}^{\perp} = \frac{\partial \mathbf{f}^{\perp}}{\partial \boldsymbol{\zeta}}\mathbf{J}_l^{\perp}\mathrm{d}\boldsymbol{\zeta}_f + \frac{\partial \mathbf{f}^{\perp}}{\partial \boldsymbol{\zeta}}\mathbf{J}_l^{\parallel}\mathrm{d}\mathbf{l} = \mathbf{0}_{\lambda\times 1} \tag{2.89}$$

Accounting for the right-hand side of Eq. (2.68) yields:

$$\frac{\partial \mathbf{f}^{\perp}}{\partial \boldsymbol{\zeta}} = -\left(\frac{\partial {\Xi_l^{\perp}}^T}{\partial \boldsymbol{\zeta}}\mathbf{f} + {\Xi_l^{\perp}}^T\frac{\partial \mathbf{f}}{\partial \boldsymbol{\zeta}}\right) \tag{2.90}$$

Since we are interested in variations around an equilibrium, we may use $\mathbf{f} = \Xi\boldsymbol{\tau}$ from Eq. (2.82) so that:

$$\frac{\partial \mathbf{f}^{\perp}}{\partial \boldsymbol{\zeta}} = -\left(\frac{\partial {\Xi_l^{\perp}}^T}{\partial \boldsymbol{\zeta}}\Xi\boldsymbol{\tau} + {\Xi_l^{\perp}}^T\frac{\partial \mathbf{f}}{\partial \boldsymbol{\zeta}}\right) \tag{2.91}$$

Differentiating ${\Xi_l^{\perp}}^T\Xi_l = \mathbf{0}_{\lambda\times n}$ with respect to $\boldsymbol{\zeta}$ yields:

$$\frac{\partial {\Xi_l^{\perp}}^T}{\partial \boldsymbol{\zeta}}\Xi_l + {\Xi_l^{\perp}}^T\frac{\partial \Xi_l}{\partial \boldsymbol{\zeta}} = \mathbf{0}_{\lambda\times n} \tag{2.92}$$

so that Eq. (2.91) can be written as:

$$\frac{\partial \mathbf{f}^{\perp}}{\partial \boldsymbol{\zeta}} = {\Xi_l^{\perp}}^T\left(\frac{\partial \Xi_l}{\partial \boldsymbol{\zeta}}\boldsymbol{\tau} - \frac{\partial \mathbf{f}}{\partial \boldsymbol{\zeta}}\right) \tag{2.93}$$

The first term in the parentheses at the right-hand side of Eq. (2.93) is calculated by considering the right-hand side of Eq. (2.21):

$$\frac{\partial \Xi_l}{\partial \boldsymbol{\zeta}}\boldsymbol{\tau} = \sum_{i=1}^{n}\tau_i\frac{\partial \boldsymbol{\xi}_{l_i}}{\partial \boldsymbol{\zeta}} = \sum_{i=1}^{n}\tau_i\begin{bmatrix}\frac{\partial \mathbf{t}_i}{\partial \boldsymbol{\zeta}}\\ \tilde{\mathbf{a}}_i'\frac{\partial \mathbf{t}_i}{\partial \boldsymbol{\zeta}} - \tilde{\mathbf{t}}_i\frac{\partial \mathbf{a}_i'}{\partial \boldsymbol{\zeta}}\end{bmatrix} \tag{2.94}$$

It can be shown by computation (see App. A.9) that:

$$\frac{\partial \mathbf{t}_i}{\partial \boldsymbol{\zeta}} = \begin{bmatrix}\mathbf{T}_i & -\mathbf{T}_i\tilde{\mathbf{a}}_i'\end{bmatrix}\mathbf{D}, \qquad \mathbf{T}_i = \frac{\sin\psi_i\mathbf{w}_i\mathbf{w}_i^T}{\varrho_{u_i}} + \frac{\mathbf{n}_i\mathbf{n}_i^T}{\|\boldsymbol{\rho}_i\|} \tag{2.95}$$

$$\frac{\partial \mathbf{a}_i'}{\partial \boldsymbol{\zeta}} = \begin{bmatrix}\mathbf{0}_{3\times 3} & -\tilde{\mathbf{a}}_i'\end{bmatrix}\mathbf{D} \tag{2.96}$$

thus obtaining:

$$\frac{\partial \Xi_l}{\partial \boldsymbol{\zeta}}\boldsymbol{\tau} = \mathbf{K}\mathbf{D} \tag{2.97}$$

$$\mathbf{K} = \sum_{i=1}^{n} \tau_i \begin{bmatrix} \mathbf{T}_i & -\mathbf{T}_i \tilde{\mathbf{a}}_i' \\ \tilde{\mathbf{a}}_i' \mathbf{T}_i & -\tilde{\mathbf{a}}_i' \mathbf{T}_i \tilde{\mathbf{a}}_i' \end{bmatrix} + \sum_{i=1}^{n} \tau_i \begin{bmatrix} \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \tilde{\mathbf{t}}_i \tilde{\mathbf{a}}_i' \end{bmatrix} \tag{2.98}$$

In the literature, the (6×6) matrix $\mathbf{K}$ is referred to as *Geometric* [51], *Controllable* [52] or *Active* [53] *Stiffness* of the *CDPR*, because it is geometry dependent and, in over-constrained *CDPRs*, $\boldsymbol{\tau}$ can be actively controlled independently from the *EE* configuration. It should be noted that its definition is fundamentally different from the so-called *Passive Stiffness* generated by cable deformations (not considered in this thesis, since cables are modelled as rigid). In *UACDPRs*, $\mathbf{K}$ cannot be actively controlled, because $\boldsymbol{\tau}$ depends on the equilibrium configuration.

The second term in the parentheses at the right-hand side of Eq. (2.93) is calculated from Eqs. (2.64) and (2.65):

$$-\frac{\partial \mathbf{f}}{\partial \boldsymbol{\zeta}} = \mathbf{E}\mathbf{D} - \mathbf{F} \tag{2.99}$$

$$\mathbf{E} = \begin{bmatrix} \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & -\tilde{\boldsymbol{\phi}}_i \tilde{\mathbf{e}}' \end{bmatrix}, \qquad \mathbf{F} = \begin{bmatrix} \frac{\partial \boldsymbol{\phi}}{\partial \boldsymbol{\zeta}} \\ \mathbf{R}\frac{\partial^P \mathbf{e}'}{\partial \boldsymbol{\zeta}} + \tilde{\mathbf{e}}' \frac{\partial \boldsymbol{\phi}}{\partial \boldsymbol{\zeta}} + \frac{\partial \boldsymbol{\mu}}{\partial \boldsymbol{\zeta}} \end{bmatrix} \tag{2.100}$$

since $\partial \mathbf{e}' / \partial \boldsymbol{\zeta} = [\mathbf{0}_{3\times3} \ -\tilde{\mathbf{e}}_i' \mathbf{D} + \mathbf{R}\frac{\partial^P \mathbf{e}'}{\partial \boldsymbol{\zeta}}]$.

Finally, substituting Eqs. (2.97) and (2.99) in (2.93), yields:

$$\frac{\partial \mathbf{f}^{\perp}}{\partial \boldsymbol{\zeta}} = \boldsymbol{\Xi}_l^{\perp^T} \left[(\mathbf{K} + \mathbf{E})\, \mathbf{D} - \mathbf{F} \right] \tag{2.101}$$

where:

$$\mathbf{K} + \mathbf{E} = \sum_{i=1}^{n} \tau_i \begin{bmatrix} \mathbf{T}_i & -\mathbf{T}_i \tilde{\mathbf{a}}_i' \\ \tilde{\mathbf{a}}_i' \mathbf{T}_i & -\tilde{\mathbf{a}}_i' \mathbf{T}_i \tilde{\mathbf{a}}_i' \end{bmatrix} + \sum_{i=1}^{n} \tau_i \begin{bmatrix} \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & -\tilde{\mathbf{t}}_i \tilde{\mathbf{a}}_i' \end{bmatrix} + \begin{bmatrix} \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \tilde{\boldsymbol{\phi}}_i \tilde{\mathbf{e}}_i' \end{bmatrix} \tag{2.102}$$

Notice that matrix $\mathbf{K} + \mathbf{E}$ is generally non-symmetric, since, while the first summation in Eq. (2.102) is always symmetric, the other terms are not. In fact, at the static equilibrium:

$$\sum_{i=1}^{n} \tau_i \mathbf{a}_i' \times \mathbf{t}_i = \mathbf{e}_i' \times \boldsymbol{\phi} + \boldsymbol{\mu} \tag{2.103}$$

namely, in skew-symmetric representation:

$$\sum_{i=1}^{n} \tau_i \left(\tilde{\mathbf{a}}_i' \tilde{\mathbf{t}}_i - \tilde{\mathbf{t}}_i \tilde{\mathbf{a}}_i' \right) = \tilde{\mathbf{e}}' \tilde{\boldsymbol{\phi}} - \tilde{\boldsymbol{\phi}} \tilde{\mathbf{e}}' + \tilde{\boldsymbol{\mu}} \tag{2.104}$$

Equation (2.104) shows that the summation of the second and third term in Eq. (2.102), namely,

$$\tilde{\boldsymbol{\phi}} \tilde{\mathbf{e}}' - \sum_{i=1}^{n} \tau_i \tilde{\mathbf{t}}_i \tilde{\mathbf{a}}_i' = \tilde{\mathbf{e}}' \tilde{\boldsymbol{\phi}} - \sum_{i=1}^{n} \tau_i \tilde{\mathbf{a}}_i' \tilde{\mathbf{t}}_i + \tilde{\boldsymbol{\mu}} = \left(\tilde{\boldsymbol{\phi}} \tilde{\mathbf{e}}' - \sum_{i=1}^{n} \tau_i \tilde{\mathbf{t}}_i \tilde{\mathbf{a}}_i' \right)^T + \tilde{\boldsymbol{\mu}} \tag{2.105}$$

is symmetric if and only if $\boldsymbol{\mu} = \mathbf{0}_{3\times1}$.

If we define:

$$\mathbf{K}_f^{\perp} = \frac{\partial \mathbf{f}^{\perp}}{\partial \boldsymbol{\zeta}} \mathbf{J}_l^{\perp} = \boldsymbol{\Xi}_l^{\perp^T} \left[(\mathbf{K} + \mathbf{E})\, \mathbf{D} - \mathbf{F} \right] \mathbf{J}_l^{\perp} \tag{2.106}$$

$$\mathbf{K}_l^{\perp} = \frac{\partial \mathbf{f}^{\perp}}{\partial \boldsymbol{\zeta}} \mathbf{J}_l^{\parallel} = \boldsymbol{\Xi}_l^{\perp^T} \left[(\mathbf{K} + \mathbf{E})\, \mathbf{D} - \mathbf{F} \right] \mathbf{J}_l^{\parallel} \tag{2.107}$$

equation (2.89) may be written as:

$$\mathbf{K}_f^{\perp} \mathrm{d}\boldsymbol{\zeta}_f + \mathbf{K}_l^{\perp} \mathrm{d}\mathbf{l} = \mathbf{0}_{\lambda \times 1} \tag{2.108}$$

In case equilibrium is altered while cables are not varying their lengths ($\mathrm{d}\mathbf{l} = \mathbf{0}_{n\times 1}$), i.e. because of the temporary variation of the external wrench, the restoring action that pushes the system back towards equilibrium generate the *Free-Motion Stiffness* (*FMS*) $\mathbf{K}_f^{\perp} = \partial \mathbf{f}^{\perp} / \partial \boldsymbol{\zeta}_f \in \mathbb{R}^{\lambda \times \lambda}$ of the *UACDPR*. The *FMS* was implicitly formulated for UACDPRs in [15], under the assumptions that cables exit the frame through eyelets, and the platform is subject to the gravitational action only. Here, we re-formulated $\mathbf{K}_f^{\perp}$ in a more general case, namely introducing the pulley geometric model and considering a generic external wrench $\mathbf{f}$.

It should be noted that, in case external loads are conservative, an equilibrium is stable if and only if the eigenvalues of $\mathbf{K}_f^{\perp}$ are real positive numbers, otherwise it is unstable. This assertion is equivalent to requiring $\mathbf{K}_f^{\perp}$ to be positive-definite [15]. In case external loads are not conservative, the analysis of $\mathbf{K}_f^{\perp}$ is not sufficient to infer equilibrium stability, and the inertial properties of the system are to be considered as well [54]. This case will be analyzed in Sec. 4.1.

When alternatively the equilibrium is altered because of a variation of the cable lengths, as in the case of small-amplitude controlled motions or cable-length control errors, a new equilibrium configuration is reached, and cable tensions vary in order to keep the *EE* in balance.

If the cable-length change is quasi-static and infinitesimal, cable tension variation can be asserted by computing the *Controlled-Motion Stiffness* (*CMS*) $\mathbf{K}^{\parallel} = \partial \boldsymbol{\tau} / \partial \mathbf{l} \in \mathbb{R}^{n\times n}$ of the *UACDPR*, which correlates small actuator displacements to a small variation of cable tension. A change in cable lenghts affects the free coordinates of the platform according to Eq. (2.108). Matrix $\mathbf{K}_f^{\perp}$ is invertible if it is not semi-definite (thus it is invertible if loads are conservative and the equilibrium configuration stable) and the variation of the free-coordinates upon a variation of the cable lengths can be evaluated as:

$$\mathrm{d}\boldsymbol{\zeta}_f = -\mathbf{K}_f^{-\perp} \mathbf{K}_l^{\perp} \mathrm{d}\mathbf{l} \tag{2.109}$$

Consequently, we can substitute Eq. (2.109) in Eq. (2.88) obtaining:

$$\mathrm{d}\boldsymbol{\zeta} = \left(-\mathbf{J}_l^{\perp} \mathbf{K}_f^{-\perp} \mathbf{K}_l^{\perp} + \mathbf{J}_l^{\parallel} \right) \mathrm{d}\mathbf{l} \tag{2.110}$$

Then, we may compute $\mathbf{K}^{\parallel}$ as:

$$\mathbf{K}^{\parallel} = \frac{\partial \boldsymbol{\tau}}{\partial \mathbf{l}} = \frac{\partial \mathbf{f}^{\parallel}}{\partial \mathbf{l}} = \frac{\partial \mathbf{f}^{\parallel}}{\partial \boldsymbol{\zeta}} \frac{\partial \boldsymbol{\zeta}}{\partial \mathbf{l}} = \frac{\partial \mathbf{f}^{\parallel}}{\partial \boldsymbol{\zeta}} \left(-\mathbf{J}_l^{\perp} \mathbf{K}_f^{-\perp} \mathbf{K}_l^{\perp} + \mathbf{J}_l^{\parallel} \right) \tag{2.111}$$

Matrix $\partial \mathbf{f}^{\parallel} / \partial \boldsymbol{\zeta}$ is evaluated similarly to $\partial \mathbf{f}^{\perp} / \partial \boldsymbol{\zeta}$ from Eq.(2.90) and the procedure is recalled hereafter. Accounting for the right-hand side of Eq. (2.75), we have:

$$\frac{\partial \mathbf{f}^{\parallel}}{\partial \boldsymbol{\zeta}} = \frac{\partial \boldsymbol{\Xi}_l^{\parallel^T}}{\partial \boldsymbol{\zeta}} \mathbf{f} + \boldsymbol{\Xi}_l^{\parallel^T} \frac{\partial \mathbf{f}}{\partial \boldsymbol{\zeta}} \tag{2.112}$$

and, around equilibrium, $\mathbf{f} = \boldsymbol{\Xi} \boldsymbol{\tau}$, thus:

$$\frac{\partial \mathbf{f}^{\parallel}}{\partial \boldsymbol{\zeta}} = \frac{\partial \boldsymbol{\Xi}_l^{\parallel^T}}{\partial \boldsymbol{\zeta}} \boldsymbol{\Xi} \boldsymbol{\tau} + \boldsymbol{\Xi}_l^{\parallel^T} \frac{\partial \mathbf{f}}{\partial \boldsymbol{\zeta}} \tag{2.113}$$

Then, differentiating ${\Xi^{\|}}^T \Xi = \mathbf{I}_{n\times n}$ with respect to $\boldsymbol{\zeta}$ and substituting yields:

$$\frac{\partial \mathbf{f}^{\|}}{\partial \boldsymbol{\zeta}} = -{\Xi_l^{\|}}^T \left(\frac{\partial \Xi_l}{\partial \boldsymbol{\zeta}} \boldsymbol{\tau} - \frac{\partial \mathbf{f}}{\partial \boldsymbol{\zeta}} \right) \tag{2.114}$$

where the terms inside the parentheses have already been calculated in Eq. (2.94) through Eq. (2.100) as:

$$\frac{\partial \Xi_l}{\partial \boldsymbol{\zeta}} \boldsymbol{\tau} - \frac{\partial \mathbf{f}}{\partial \boldsymbol{\zeta}} = (\mathbf{K} + \mathbf{E})\,\mathbf{D} - \mathbf{F} \tag{2.115}$$

Finally, we have:

$$\mathbf{K}^{\|} = -{\Xi_l^{\|}}^T \left[(\mathbf{K} + \mathbf{E})\,\mathbf{D} - \mathbf{F} \right] \left(-\mathbf{J}_l^{\perp} \mathbf{K}_f^{-\perp} \mathbf{K}_l^{\perp} + \mathbf{J}_l^{\|} \right) \tag{2.116}$$

In the end, if the force $\boldsymbol{\phi}$ is constant and $\boldsymbol{\mu} = \mathbf{0}_{3\times 1}$, then $\mathbf{F} = \mathbf{0}_{6\times 6}$, and both the *FMS* and *CMS* are symmetric for *any* choice of *EE* pose parameters (both $\mathbf{p}$ and $\boldsymbol{\epsilon}$):

$$\mathbf{K}_f^{\perp} = {\Xi_l^{\perp}}^T (\mathbf{K} + \mathbf{E})\, \Xi_l^{\perp} \tag{2.117}$$

$$\mathbf{K}^{\|} = -{\Xi_l^{\|}}^T (\mathbf{K} + \mathbf{E}) \left(-\Xi_l^{\perp} \mathbf{K}_f^{-\perp} {\Xi_l^{\perp}}^T (\mathbf{K} + \mathbf{E}) + \mathbf{I}_{6\times 6} \right) \Xi_l^{\|} \tag{2.118}$$

Chapter 3

Static Workspace Characterization

The workspace of a robotic manipulator can be defined as *the set of configurations* $\boldsymbol{\zeta}$ *that can be reached by its* EE. It is a crucial property that defines the robot possible applications. This set is commonly limited by different factors, such as mechanical limits on passive joints, or self-collision between the elements of the robot. An additional factor which is specific to any cable robot is that cable tension must always vary in a prescribed non-negative range.

This chapter aims at summarizing commonly employed definitions for *CDPR* workspaces, and their adaptation to *UACDPRs*. For this class of manipulators, workspace calculation relies on the evaluation of mechanical equilibrium as well as geometrical considerations. A novel performance indexes which is specific to *UACDPRs* is proposed and a novel fast workspace computation algorithm is detailed. Examples on 2-, 3-, and 4-cable robots are analyzed.

3.1 Common *CDPR WS*

In the following, commonly used workspace definitions in the context of parallel robots are reported (see also [55]):

- *constant-orientation workspace or translation workspace*: all possible locations of the reference point $\mathbf{p}$ of the *EE* that can be reached with a given orientation $\boldsymbol{\epsilon}$;
- *orientation workspace*: all possible orientations $\boldsymbol{\epsilon}$ that can be reached while the reference point $\mathbf{p}$ is in a fixed location;
- *maximal workspace or reachable workspace*: all locations of the reference point $\mathbf{p}$ that may be reached with at least one orientation $\boldsymbol{\epsilon}$ of the *EE*;
- *total orientation workspace*: all locations of the reference point $\mathbf{p}$ that may be reached with all orientations $\boldsymbol{\epsilon}$ within a set defined by ranges on the orientation angles.

There are different criteria to decide if a pose belongs to the workspace:

- passive joints usually have mechanical limits that cannot be hit or surpassed, because of a consequent robot-hardware failure;
- elements of the robot may self-collide (cable-cable or cable-platform), causing the robot geometric and mechanical model to vary w.r.t. the nominal one;

- actuators usually have a limited stroke, thus a mechanical limit: this is usually neglected in cable-robotics, since the possibility to have practically unlimited strokes is one of the primary advantages of *CDPR* w.r.t. conventional parallel robots;

- robot legs reach singular configurations.

Cable interference criteria led to the definition of the *Interference-Free* Workspace [56–58], where the platform can safely reach configurations without its cables interfering with themselves or specific elements in the environment. Although cable collision implies loss of accuracy, the flexible nature of this transmission may not necessarily threaten the robot integrity [59]. This aspect, as well as passive joints mechanical limits, has limited impact for the suspended *UACDPRs* considered in this thesis, since in this case the possibility of cable self-collision is limited, if not completely avoided, by the suspended configuration and by the presence of a small number of cables.

Probably, the most important criterion that is specific to cable robots, though, is that cables can only be actuated unilaterally through tension and not compression. This consideration led to the definition of an additional specification to traditional workspaces, which is the so-called wrench-closure property of *CDPR* workspaces [60, 61]: the set of configurations belonging to a *Wrench-Closure* workspace can be statically maintained by cables exerting a non-zero positive tension when a given external load (often gravitational only) is acting upon the *EE*. More realistically, if cable tensions must be limited within lower and upper positive bounds, the workspace is called *Wrench-Feasible* (*WFW*) [62].

Since mechanical equilibrium plays a key role in *CDPR* workspace definitions, it is commonly assumed that cable tensions must counteract external static loads only, so that *EE* is able to reach a configuration belonging to the workspace at rest [63, 64]. On the other hand, recent studies also focused on the ability of *CDPRs* to move outside of their static workspace when they are not in a static configuration, thus defining the concept of *dynamic* workspace [65].

To the knowledge of the author, there exists a single work addressing the determination of the workspace of *UACDPRs*. The authors of [66] address this problem as the *determination of the set of poses* $\boldsymbol{\zeta}$ *that a* 6-DoF EE *constrained by 4 cables can attain statically, by taking only gravity into account.* This set is determined by discretizing the workspace and solving a nonlinear optimization problem with static equilibrium conditions as the objective function, and constraints on the platform orientation and cable tension values. While the approach presented in [66] appears simple and effective enough, several details regarding the computation method are missing or obscure (even though the platform moves in $SE(3)$, authors only mention an "angle", without parametrization specifications, that must vary in a prescribed range).

In the following, a novel performance index *UACDPRs*, the *maximum tension variation under a unit-norm cable displacement*, is introduced. Accounting for this index, a novel workspace definition, the *Tension-Error-Insensitive Workspace* (*TEIW*), is put forward, and a computation algorithm detailed. Exemplary applications on 2-, 3-, and 4-cable *UACDPR* will show how this index is particularly suitable for characterizing manipulators with non-decoupable translational and rotational mechanical equilibrium.

3.2 *Tension-Error-Insensitive* workspace

Usually, kinematic performance indices are used in order to compare robot architectures, and drive mechanical and control design. For example, it is often desired to estimate the local kinematic sensitivity of a robotic manipulator, i.e., in a given posture, measure the effect of actuator displacements on the displacements of its end-effector [67, 68]. In this Section, a novel performance index is proposed, which is tailored for *suspended CDPRs*, thus applicable to the *UACDPRs* considered in this thesis. In fact, it is based on three critical aspects of this technology:

- cable tension needs to be strictly bounded within positive limits for safety and controllability reasons;
- cable-length estimation may not be very precise if fiber-rope cables or complex cable-routing devices are employed;
- cable tension cannot be adjusted and depends on the *EE* pose.

Thus, it makes sense to investigate the sensitivity of robot cable tensions under errors in cable-length assignments, namely, in other words, to identify *CDPR* configurations which are more robust in terms of maintaining cables under tension when cable lengths are not precisely estimated.

This index can be formulated by recalling the results of Section 2.4.2, which led to the definition of the *CMS* of *UACDPR* $\mathbf{K}^{\|}$. In fact, matrix $\mathbf{K}^{\|}$ correlates small actuator displacements (or errors) to small variation of cable tensions for *UACDPRs* with *non-deformable* cables:

$$\mathrm{d}\boldsymbol{\tau} = \mathbf{K}^{\|}\mathrm{d}\mathbf{l} \tag{3.1}$$

This definition can be used to compute a novel geometrico-static performance index for *UACDPR*: *the maximum tension variation under a unit-norm cable displacement*:

$$\sigma_{\tau,q} = \max_{\|\mathrm{d}\mathbf{l}\|_q=1} \|\mathrm{d}\boldsymbol{\tau}\|_q \tag{3.2}$$

where $\|\cdot\|_q$ indicates the vector q-norm. As it was highlighted in [69], and confirmed in [67], the use of the infinity-norm ($q=\infty$) attains the clearest physical meaning, since it is consistent with a realistic actuation error model, that is:

$$-\mathrm{d}l_{i,\max} \le \mathrm{d}l_i \le \mathrm{d}l_{i,\max}, \qquad i = 1,\dots,n \tag{3.3}$$

Thus, by considering Eq. (3.1) and the definitions of matrix norms [49], we have:

$$\sigma_{\tau,\infty} = \max_{\|\mathrm{d}\mathbf{l}\|_\infty=1} \|\mathrm{d}\boldsymbol{\tau}\|_\infty = \|\mathbf{K}^{\|}\|_\infty \tag{3.4}$$

Note that the computation of the infinity norm of a matrix is straightforward, as it consists of computing the 1-norms of its row vectors and choosing the largest.

According to the index value over the whole *WFW*, comparisons between different design choices can be made: the smaller the index, the more robust to error in cable actuation a *UACDPR* is. Alternatively, if the design is fixed, various robot configurations can be compared:

- while planning a trajectory, some configurations may be avoided because they are not safe to pass through;

- in some robot architectures, e.g. the 4-cable robot, if a positioning task is to be performed, orientation parameters can be optimized so as to minimize $\sigma_{\tau,\infty}$ while following a trajectory of the *EE* reference point.

Additionally, if clear limits on cable errors are known, one may restrict the workspace to those poses which are intrinsically *"safe"* to reach even in case of actuator maximal errors, since cables will most likely stay within their non-zero positive bounds. Upper $\boldsymbol{\tau}_u = [\tau_{u,1} \ldots, \tau_{u,n}]^T$ and lower $\boldsymbol{\tau}_l = [\tau_{l,1} \ldots, \tau_{l,n}]^T$ limits od cable tensions in case actuator present a maximum error of magnitude Δl can be computed according to:

$$\tau_{l,i} = \tau_i - \|\mathbf{K}_i^{\parallel}\|_1 \Delta l, \qquad \tau_{u,i} = \tau_i + \|\mathbf{K}_i^{\parallel}\|_1 \Delta l, \qquad i = 1, \ldots, n \tag{3.5}$$

where $\|\mathbf{K}_i^{\parallel}\|_1$ is the 1-norm of $\mathbf{K}^{\parallel}$ i-th row and τ_i is the i-th cable tension as calculated in Eq. (2.84). Obviously, $\sigma_{\tau,\infty} = \max\|\mathbf{K}_i^{\parallel}\|_1$ for $i = 1, \ldots, n$, but the maximum value of this norm may not be the one which makes a cable tension surpass its bounds for a given Δl, thus each row needs to be considered. In the end, cable tension bounds used for the *wrench-feasibility* check can be compared with $\tau_{l,i}$ and $\tau_{u,i}$ instead of τ_i, and, if a pose is wrench feasible for $\tau_{l,i}$ and $\tau_{u,i}$, it is said to be *Tension-Error-Insensitive*.

3.3 Algorithm for Reachable Workspace Computation

The definition of a *wrench-feasible reachable workspace* is sound for any number of cables:

- in case $n = 2$, if two translational components of the *EE* pose $\boldsymbol{\zeta}$ are controlled, this kind of workspace defines a surface in $\mathbb{R}^3$[1]: surface point coordinates are given by the controlled coordinates and the free translational coordinate resulting from *EE* static equilibrium. Orientation parameters are determined according to the *EE* static equilibrium as well;

- in case $3 \leq n \leq 5$, a volume in $\mathbb{R}^3$ is defined, by controlling at least the 3 translational coordinates of $\boldsymbol{\zeta}$. For $4 \leq n \leq 5$, some orientation coordinates can be controlled as well, thus the *EE* may attain several orientation configurations for a fixed reference point;

- tension limits on every cable are easily verifiable.

It should be noted that, in some very specific but practically relevant cases, such as cables exiting the frame through eyelets and external loads being the gravitational one only, the surfaces defined by the *wrench-feasible reachable workspace* of a 2-cable robot is a plane.

Numerous papers dealing with workspace calculation use methods based on the discretisation of the pose parameters, in order to determine the workspace boundary. In this discretisation approach, the workspace is covered by a regular or adaptive grid, either Cartesian or polar, of nodes. Each node is then tested to see whether it belongs to the workspace, according to various criteria. The main advantages of this approach are:

- workspace calculation algorithms are straightforward;

[1] The surface shape is in general unknown a-priori, but it is *practically* a plane for limited pulley radius

- any criteria is easily included;
- it is possible to additionally evaluate performance indexes associated with workspace nodes.

On the other hands, the most noticeable drawbacks are:

- the accuracy of the workspace boundary depends on the grid resolution, and the computation time grows exponentially with the resolution;
- problems may occur when the workspace possesses voids;
- the boundary representation may involve a large number of nodes.

In the following, our aim is not the accurate determination of workspace boundaries, but the rapid determination of a finite set of configurations that the *EE* is able to reach (i) without the need of re-assembly or crossing singularities of any kind, (ii) while statically maintaining cable tension within given bounds: thus, a regular-grid discretization method is employed for workspace computation. The main steps for workspace calculation are summarized as follows:

1. upper and lower limits of controlled pose coordinates, namely $\boldsymbol{\zeta}_{c,u}$ and $\boldsymbol{\zeta}_{c,l}$ respectively, are assigned;
2. the intervals defined by these limitis are divided by a regular grid of n_g nodes, where n_g is odd (thus there is a central node in each interval); the number of nodes n_g is determined by means of geometric considerations on the robot footprint;
3. for each controlled-pose node, a solution of the *IGSP* in Eq. (2.85) is sought; if a solution for the free pose coordinates is found, cable tensions are checked according to Eq. (2.84);
4. In case cable tensions are positive, that is, the pose belongs to the *wrench-closure* workspace, additionally restrictive criteria are checked, namely:
   - cable tensions are checked against their lower and upper positive bounds, so that the pose is included in the *wrench-feasible* workspace;
   - if the pose is wrench-feasible, equilibrium stability is checked, according to the results of Sec. 2.4.2;
5. if all these additional restrictions apply, cable maximum error Δl is set, and $\boldsymbol{\tau}_u$ and $\boldsymbol{\tau}_l$ are computed according to Eq. (3.5); tension lower and upper positive bounds are checked in order to include the pose in the *Tension-Error-insensitive* workspace (*TEIW* in short).

One of the key complexities of the proposed steps is the solution of the *IGSP*, which is nonlinear in $\boldsymbol{\zeta}_f$. If a solution is sought by using a numerical iterative solver, such as a Newton-Raphson scheme, the choice of an appropriate initial guess for the free pose coordinates is fundamental. Such an initial guess should satisfy the Kantorovich theorem [34], which provides sufficient conditions for the existence and uniqueness of a solution to a nonlinear system of equation, given a specific initial guess. Algorithm 1 proposes an euristic, which allows for:

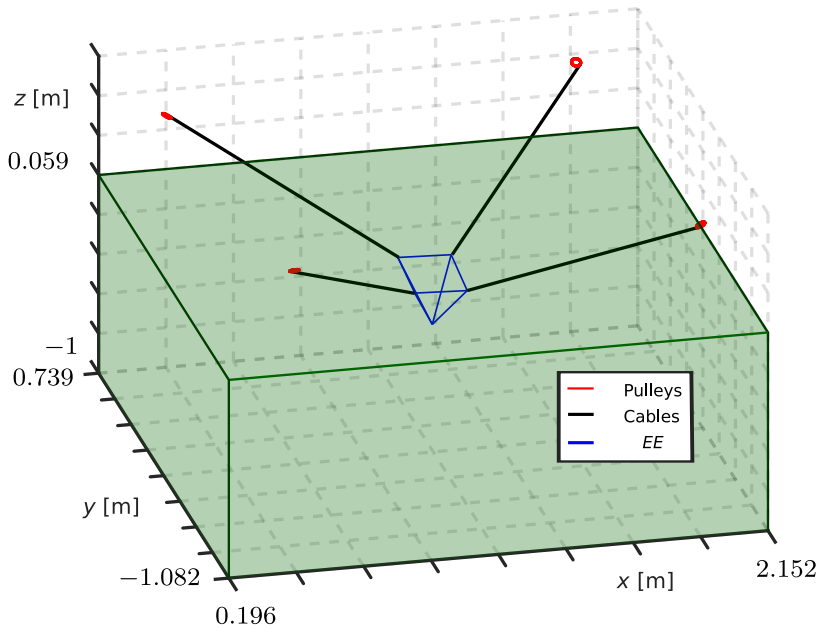


Figure 3.1: Example of a 4-cable suspended *UACDPR* with an assigned installation volume: the volume (in green) is conveniently reduced in the z direction with respect to the z coordinates of points D_i in order to decrease computation time.

- fast *UACDPRs* workspace computation for any number of cables within 2 and 5 ($2 \leq n \leq 5$);
- the assignment of an initial guess for the *IGSP* which is most likely to be compatible with the Kantorovich theorem;
- each configuration is checked only one time because of the algorithm structure.

In the case the *EE* is suspended and it is supposed to move inside a volume defined by an installation frame and possibly other space constraints (see Fig. 3.1), points $\mathbf{d}_i$ are on the volume boundaries and their z coordinates are on the volume upper part (or over the upper part, such as in Fig. 3.1). The algorithm requires to choose as many position coordinates as possible as controlled coordinates (thus 2 position coordinates for a 2-cable robot and 3 position coordinates for 3-, 4-, and 5-cable robots): their limits can be easily chosen by considering the robot installation volume. When additional orientation coordinates can be selected as controlled (in the 4- and 5-cable cases), it is useful that their limits have zero mean. The algorithm then starts from the center of the controlled coordinate intervals and sequentially increases or decreases one of the controlled coordinates. This choices lead to a fairly simple heuristic selection of the very first initial guess for the numerical solution of the *IGSP*, if only gravity is considered as an external load for workspace computation:

- in case the 2-cable robot is considered, the free position coordinate initial value is selected as the corresponding coordinate in the center of the robot installation volume; orientation free coordinates are initialized to the zero vector of dimension 3, which corresponds to the *EE* z' axis pointing upwards, thus parallel to the fixed vertical z axis;
- for the 3-, 4-, and 5-cable robot, the orientation free coordinates are initialized to 0, which again corresponds to the *EE* z' axis pointing upwards.

Algorithm 1: Workspace Computation

Data: $\boldsymbol{\zeta}_{c,l}$, $\boldsymbol{\zeta}_{c,u}$, τ_l, τ_u, Δl, n_g, *UACDPR* geometry and inertial parameters
Result: Set of configurations $\boldsymbol{\zeta}$ belonging to the *WFW*

```
c_c = (ζ_{c,u} + ζ_{c,l})/2                          /* Controlled coordinates center */
ζ_c = c_c                                            /* Initialize controlled coordinates */
s_n = (n_g − 1)/2                                    /* Nodes on each side of the center */
Γ = 0_{λ,n}                                          /* Matrix of auxiliary initial guesses for the IGSP */
γ = Γ(:, n)                                          /* Initial guess for the IGSP */
δζ_c = (ζ_{c,u} + ζ_{c,l})/(n_g − 1)                 /* Constant distances between nodes */
iter = 0_n,    iter(n) = −1
dir = 1_n
i = n
while 1 do                                           /* Computation starts */
    iter(i) = iter(i) + 1
    ζ_c(i) = c_c(i) + dir(i)iter(i)δζ_c(i)           /* One coordinate is updated */
    if iter(i) > s_n then                            /* limit is reached for i-th coordinate */
        γ = Γ(:, i)
        if dir(i) = 1 then                           /* Limit is upper */
            dir(i) = −1
            iter(i) = 0
        else                                         /* Limit is lower */
            dir(i) = 1
            iter(i) = −1
            i = i − 1                                /* Decrease the coordinate to update */
            if i = 0 then
                Exit While                           /* Computation is finished */
            end
        end
    else
        if i = n then
            ζ_f ← Solve IGSP for ζ_c, starting from γ
            γ = ζ_f
            ζ = P^T [ζ_c^T, ζ_f^T]^T
            τ = f^∥
            if τ is within limits and ζ is stable then
                Assign ζ to WFW
                Compute σ_{τ,∞}
                Compute τ_l and τ_u
                if τ_l and τ_u are within limits then
                    Assign ζ to TEIW
                end
            end
            if iter(n) = 0  &  dir(n) = 1 then
                Γ(:, i) = γ      /* Update auxiliary matrix of initial guesses when
                  coordinate n near is near the center */
            end
        else
            if iter(i) = 1  &  dir(i) = 1 then
                Γ(:, i) = Γ(:, i + 1)    /* Update auxiliary matrix of initial guesses
                  when coordinate i is near the center */
            end
            i = i + 1
                                             /* Increase the coordinate to update */
        end
    end
end
```

In this way, *IGSP* numerical solution is forced towards the one (among the possibly many) solution with the *EE* which is minimally tilted with respect to the fixed vertical z axis: it is not the aim of this algorithm to find all the solutions to the *IGSP* problem, but rather to determine a limited and finite set of configurations that the *EE* can practically reach during operation. It should be noted that, in case different geometrical assumptions and external loads are considered, the choice on controlled coordinate limits and free coordinate initial guess would probably slightly vary.

After the first node, the initial guesses for *IGSP* solution is always selected as the solution obtained in the previous node, in order to start from the solution of a neighbouring problem. Each time an upper or lower limit is reached while increasing or decreasing controlled coordinates, the algorithm goes back to a configuration near the center, and the initial guess is accordingly updated with the solution to the nearest problem already solved.

3.3.1 Results on exemplary 2-, 3- and 4-cable robots

Workspaces for exemplary 2-, 3-, and 4-cable robots are evaluated according to Alg. 1 in order to show its effectiveness. Geometrical and inertial parameters of the robot *EE* are summarized in App. B.1, in tables B.1 and B.2 respectively. Apart from the 4-cable case, where all cables were connected to the *EE*, only cables 1 through 3 were attached to the platform in the 3-cable case, only cables 1 and 3 in the 2-cable robot.

Pose limits were set to (see Fig. 3.1):

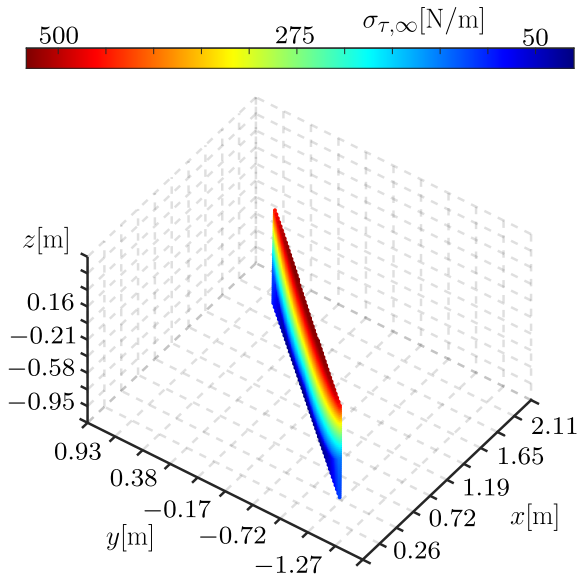
$$\boldsymbol{\zeta}_l = \begin{bmatrix} 0.196 \\ -1.082 \\ -1 \\ -\pi \\ -\pi/2 \\ -\pi/10 \end{bmatrix} \text{m, rad} \qquad \boldsymbol{\zeta}_u = \begin{bmatrix} 2.152 \\ 0.739 \\ 0.059 \\ \pi \\ \pi/2 \\ \pi/10 \end{bmatrix} \text{m, rad} \tag{3.6}$$

and cable tension limits to $\tau_l = 10$N and $\tau_u = 200$N. The position of the reference point and the *EE* torsion angle were selected as controlled coordinate of the 4-cable robot, whereas, only the position of the *EE* was chosen for the 3-cable robot, and the y, z coordinates for the 2-cable robot, thus resulting in the following permutation matrices $\mathbf{P}_n$, with n being the number of cables:

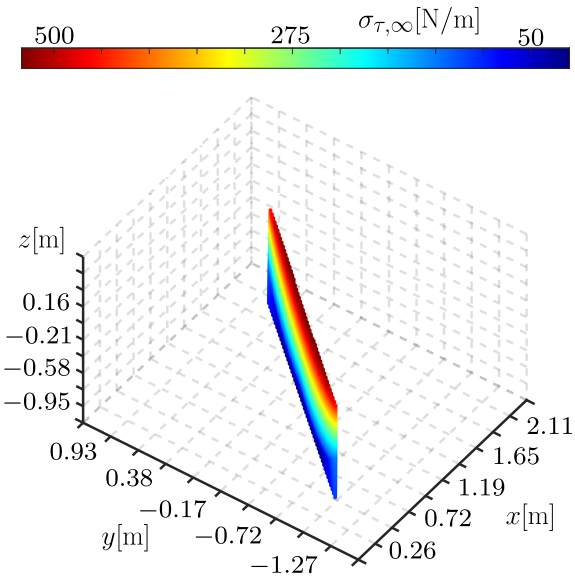
$$\mathbf{P}_4 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}, \qquad \mathbf{P}_3 = \mathbf{I}_{6\times 6}, \qquad \mathbf{P}_2 = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.7}$$

TEIWs for the 2-cable *UACDPR* are reported in Fig. 3.2. By comparing the *WFW*,namely *TEIW* with $\Delta l = 0$m in Fig. 3.2a, with *TEIWs* computed for increasing values of Δl, namely $\Delta l = 0.001$m in Fig. 3.2b, $\Delta l = 0.005$m in Fig. 3.2c, and $\Delta l = 0.01$m in Fig. 3.2d, it can be inferred that:

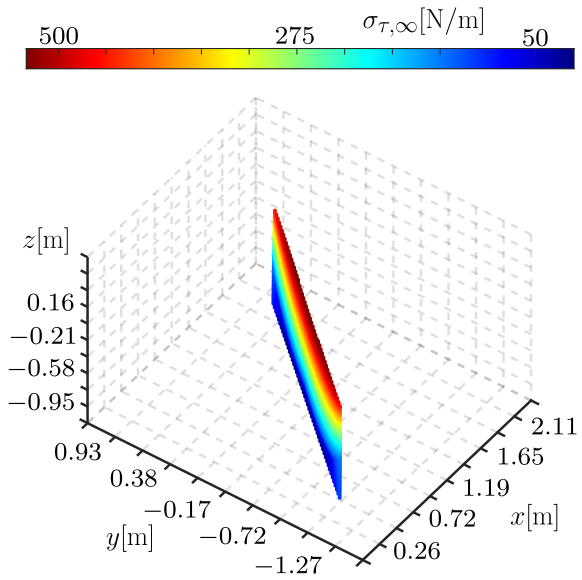
- 2-cable robots cable tensions are limitedly influenced by cable-length errors as no cable is expected to lose tension throughout robot *WFW*: *WFW* is practically coincident with *TEIWs* with Δl up to 1cm;



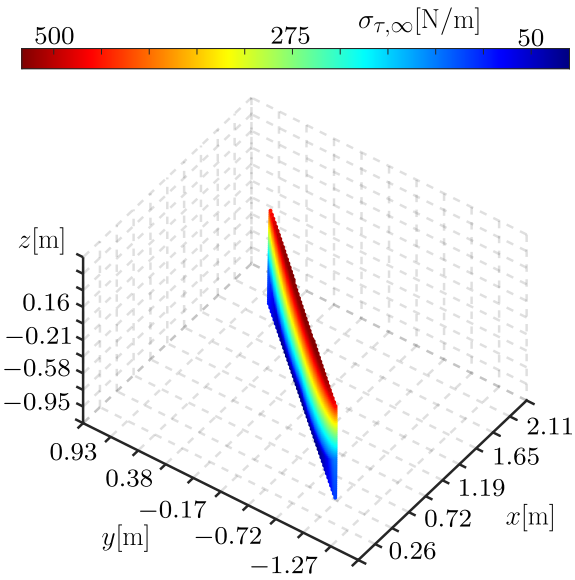
(a) 2-cable *UACDPR WFW*



(b) 2-cable *UACDPR TEIW*, $\Delta l = 1$mm



(c) 2-cable *UACDPR TEIW*, $\Delta l = 5$mm



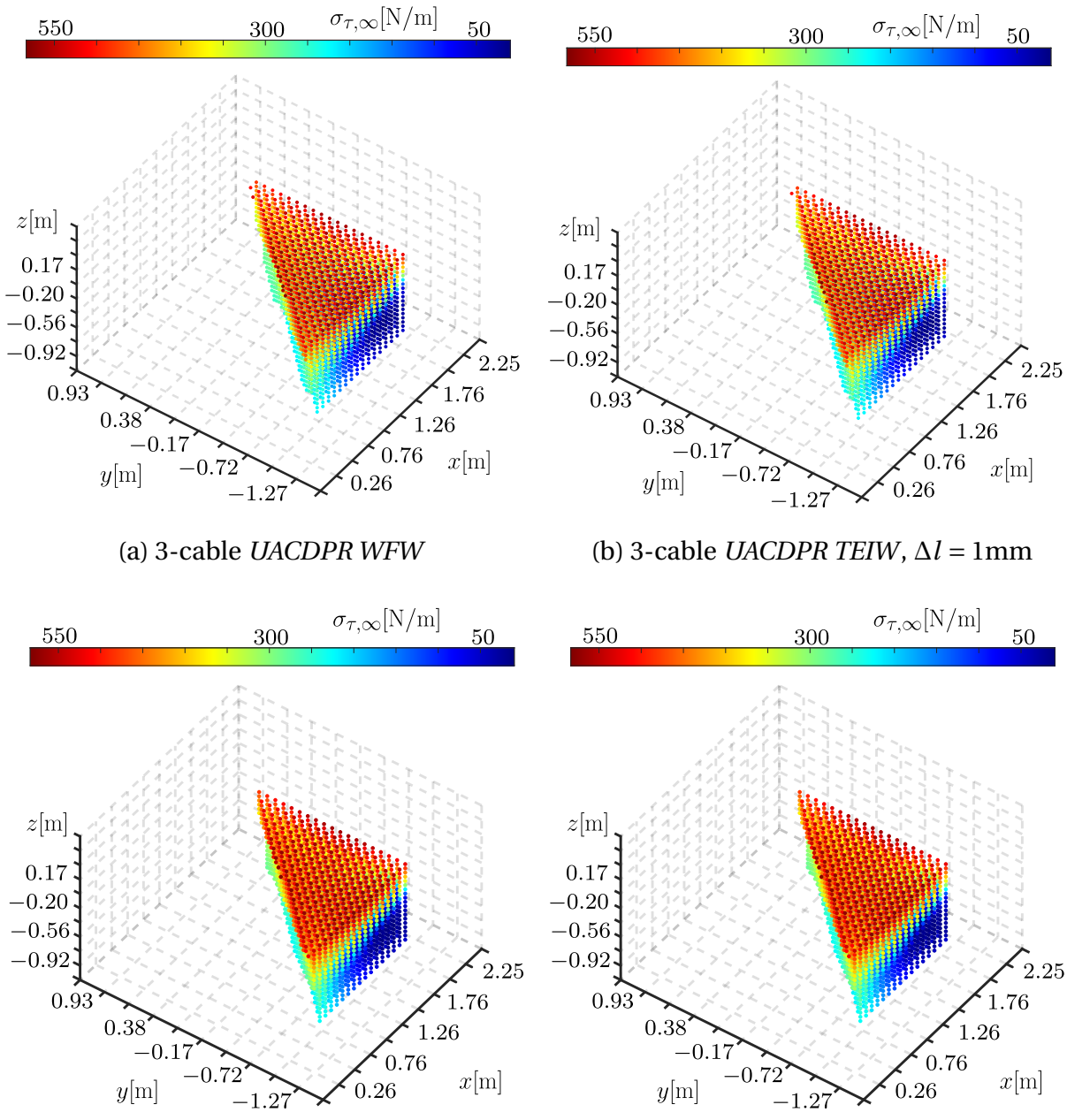
(d) 2-cable *UACDPR TEIW*, $\Delta l = 1$cm

Figure 3.2: 2-cable *UACDPR* workspaces; point colors indicate the value of $\sigma_{\tau,\infty}$ performance index; number of nodes per controlled coordinate $n_g = 101$

- the surface of the *WFW* is practically planar.

TEIWs for the 3-cable *UACDPR* are reported in Fig. 3.3. By comparing the *WFW*,namely *TEIW* with $\Delta l = 0$m in Fig. 3.3a, with *TEIWs* computed for and increasing value of Δl, namely $\Delta l = 0.001$m in Fig. 3.3b, $\Delta l = 0.005$m in Fig. 3.3c, and $\Delta l = 0.01$m in Fig. 3.3d, it can be inferred that:

- 3-cable robots cable tensions are also limitedly influenced by cable length errors as no cable is expected to lose tension throughout robot *WFW*: *WFW* is practically coincident with *TEIWs* with Δl up to 1cm;



(a) 3-cable *UACDPR WFW*

(b) 3-cable *UACDPR TEIW*, $\Delta l = 1$mm

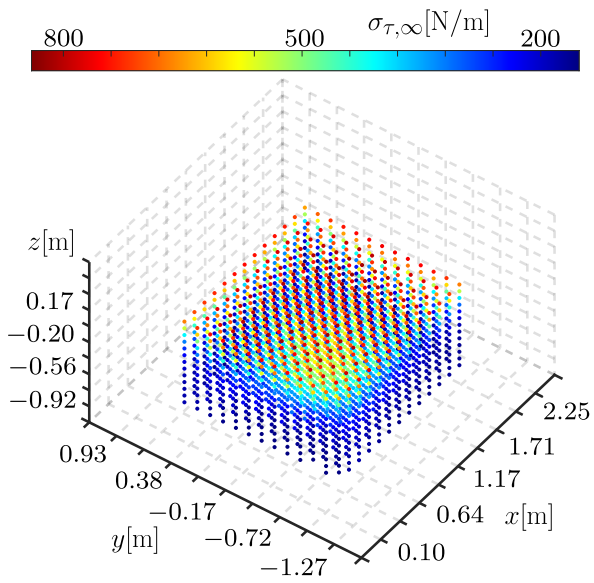
(c) 3-cable *UACDPR TEIW*, $\Delta l = 5$mm

(d) 3-cable *UACDPR TEIW*, $\Delta l = 1$cm

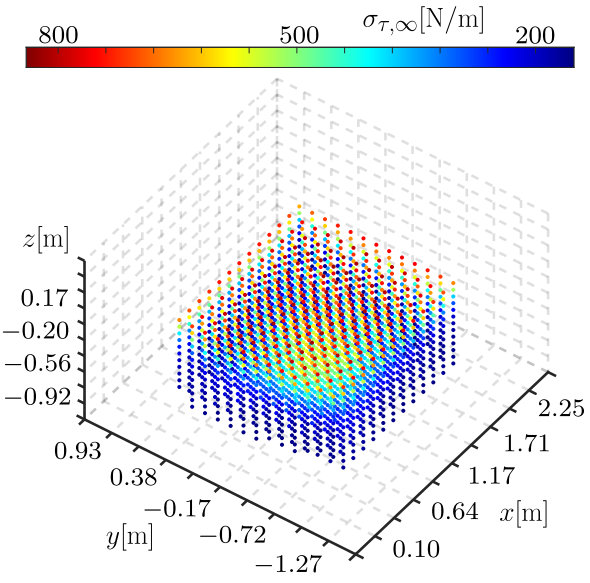
Figure 3.3: 3-cable *UACDPR* workspaces; point colors indicate the value of $\sigma_{\tau,\infty}$ performance index; number of nodes per controlled coordinate $n_g = 21$

- manipulator workspace is almost equivalent to a prism with a triangular base, which is roughly coincident with the robot installation volume: this is a favorable characteristic of the 3-cable architecture, since its *EE* can practically reach any point contained in its installation volume without cables becoming slack.

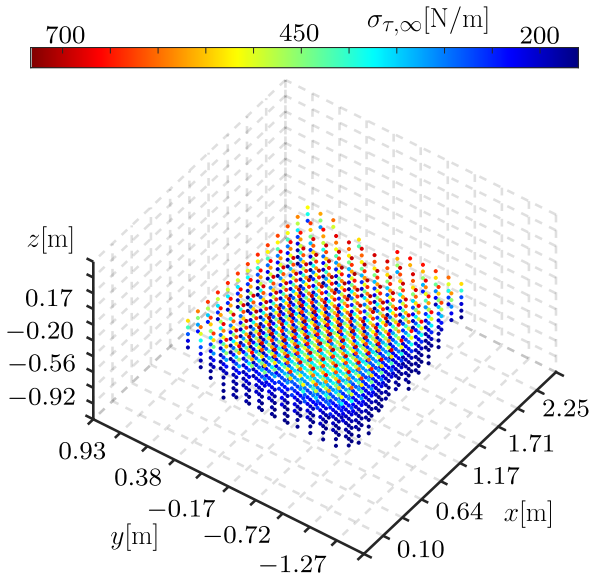
2- and 3-cable architectures are special, since translational and rotational equilibrium are decoupled, and the static problem is much simplified. In fact, it is possible to determine the full pose of the platform by considering that the two or three lines defined by the cable constraint forces shall pass through the *EE* center of mass: thus, their external moments about the center of mass are always zero.



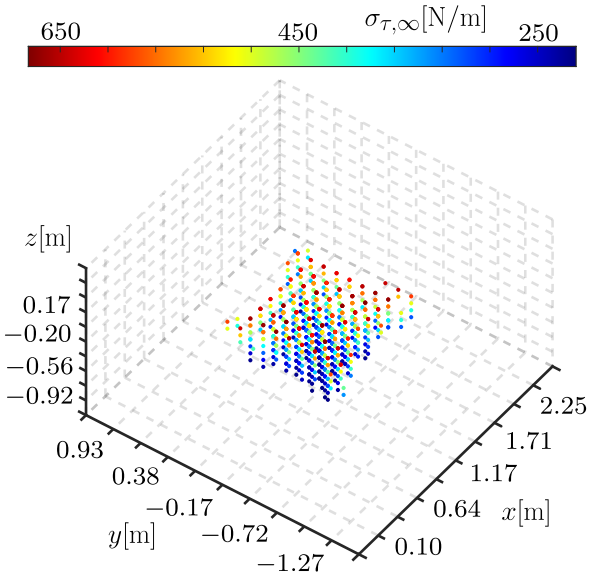
(a) 4-cable *UACDPR WFW*



(b) 4-cable *UACDPR TEIW*, $\Delta l = 1$mm



(c) 4-cable *UACDPR TEIW*, $\Delta l = 5$mm



(d) 4-cable *UACDPR TEIW*, $\Delta l = 1$cm

Figure 3.4: 4-cable *UACDPR* workspaces; point colors indicate the value of $\sigma_{\tau,\infty}$ performance index; number of nodes per controlled coordinate $n_g = 15$

The case is different for the 4-cable architecture, where there is an inherent coupling between rotational and translational equilibria. *TEIWs* for the 4-cable *UACDPR* are reported in Fig. 3.4. By comparing the *WFW*, namely *TEIW* with $\Delta l = 0$m in Fig. 3.4a, with *TEIWs* computed for and increasing value of Δl, namely $\Delta l = 0.001$m in Fig. 3.4b, $\Delta l = 0.005$m in Fig. 3.4c, and $\Delta l = 0.01$m in Fig. 3.4d, it can be inferred that:

- 4-cable robots cable tensions are severely influenced by cable-length errors and even slight actuation error (1cm on a robot with a footprint of 4m^2) may lead to one or more cables loosing tension. This result is in accordance with the findings of [33], where the *non-robust* nature of the 4-cable architecture was statistically

investigated;

- manipulator *WFW* is almost equivalent to a prism, whose edges are parallel to the external load direction (gravity): the incomplete equivalence is due to *EE* and frame generic geometry, as well as the presence of swivel pulleys.

Special care should be paid in the design and control of a 4-cable *UACDPR*, since its behaviour may largely vary with respect to expectations.

Chapter 4

Parameter Identification

Geometric and mechanical models of a robot depend on several time-invariant *parameters*, whose knowledge allows for the computation of time-varying *variables*, such as joint coordinates, *EE* pose, etc. An identification procedure is any method aiming at determining the values of said parameters in order to minimize model errors according to real-world data.

Geometric parameters, also known as *kinematic parameters*, appear in geometric and kinematic models. Their knowledge is necessary to reach a high level of motion accuracy, and they can usually be inferred with good approximation through Computer-Aided-Design (*CAD*) data. The first source of uncertainty in their knowledge is mechanical manufacturing and assembly, which result in the deviation of the real system with respect to the *CAD* one. On the other hand, robot real behaviour almost always also depends on un-modelled effects, such as link elasticity: this results in model errors even in case mechanical manufacturing and assembly are perfect. Thus, there is the need to determine (optimize) such parameters, so that the modelling errors are the smallest throughout the workspace of the manipulator. Additionally, some robot geometric parameters may inherently be impossible to know with good approximation by *CAD*: this is, for example, the case of the coordinate of points D_i in a *CDPR* (see Fig. 2.1), which can be several meters apart and may have to adapt to an unknown enviroment.

Inertial parameters only affect mechanical equilibrium and are usually needed in order to employ modern control techniques, such as computed torque control. On the other hand, Chapter 2 showed how *UACDPR* pose depends on mechanical equilibrium conditions, thus their knowledge is crucial in order to operate this class of manipulators. The knowledge of inertial parameters is often poor or even non-existent: on the one hand, data of commercial components are not always provided by manufacturers, on the other, even in-house designed components may require tedious and error-prone procedures to obtain a direct estimation.

Additional parameters, which are a by-product of geometric and inertial ones, may be useful for particular robot operations: this is the case of *UACDPR* natural oscillation frequencies, which can be very useful in the context of their trajectory planning, and *UACDPR* initial cable lengths at start-up. While the former depend on the robot parameters and the *EE* pose, thus can be modelled, the latter need to be determined each time the machine is started.

This Chapter aims at describing procedures for the determination of *UACDPR*:

- inertial parameters

- natural oscillation frequencies
- initial cable lengths

Modelling peculiarities of *UACDPRs* will be discussed, and efficient algorithms for both data acquisition and computation will be outlined. Even though natural oscillation frequency computation is effectively carried out after inertial parameters estimation, the optimal selection of configurations and trajectories for such estimations relies on the dynamic model linearization. This linearization is at the basis of the definition of natural oscillation frequencies, which are thus treated first. Unfortunately, geometric parameters calibration could not be analyzed during the time of this thesis and is left as future work.

4.1 Natural Frequencies of *UACDPR*

The knowledge of natural oscillation frequencies of *UACDPRs* was effectively employed in order to derive point-to-point trajectory planners based on periodic excitation [27] or input-shaping [28–30, 46], and it might also be exploited for optimal robot design [70]. The authors of [27, 28] derived the single configuration-dependent natural oscillation frequency of a planar 3-*DoF* 2-cable robot, by linearizing the robot dynamics about equilibrium configurations, whereas in [29] the same technique was employed for a spatial 6-*DoF* 3-cable system. In both cases, the translational and rotational equilibria of the *EE* could be decoupled, leading to equations which were mathematically simpler to handle. This was not the case, instead, for the 6-*DoF* 4-cable manipulator considered in [30]: here, the natural frequencies of the system were computed by the authors by approximating the 6-*DoF* robot with two 3-*DoF* planar systems.

In this Section, a novel technique for the computation of the natural oscillation frequencies of *UACDPR* with a generic number n of cables ($2 \leq n \leq 5$), and subject to a generic external wrench, is proposed. This method is based on the linearization of the system internal dynamics about an equilibrium configuration. The proposed approach is able to deal with the coupling of translational and rotational mechanical equilibria when $n \geq 4$, and allows out-of-the-plane oscillation frequencies of planar systems with $n = 2$ to be determined. Additionally, this method does not depend on the specific parametrization of the *EE* orientation, thus it does not suffer from representation singularities. In the end, experiments verify that the oscillation frequencies computed by the proposed approach closely match the ones of physical 2, 3 and 4-cable *UACDPR* prototypes.[1]

4.1.1 Internal-dynamics linearization

The *UACDPR* natural oscillation frequencies can be computed from the eigenvalue problem arising from the *EE* internal dynamics, after its linearization about an equilibrium configuration. For this purpose, it is convenient to linearize Eq. (2.71) by expanding it in Taylor series and truncating the expansion at the first order (an example of application to the linearization of the dynamic model of fully-actuated parallel manipulators can be found in [71]). In the following, an approach similar to [71] is followed,

[1] The content of this Section has been submitted to *IEEE Transaction on Robotics*.

but we consider $\boldsymbol{\zeta}_f$ and $\mathbf{l}$ as indipendent coordinates, instead of $\boldsymbol{\zeta}$. For simplicity sake, we define a new coordinate vector $\mathbf{q} = [\boldsymbol{\zeta}_f^T, \mathbf{l}^T]^T$.

If the left-hand side of Eq. (2.71) is denoted as $\mathbf{h}\left(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}\right)$, its Taylor expansion about an equilibrium configuration ($\mathbf{q} = \dot{\mathbf{q}}_0$, $\dot{\mathbf{q}} = \mathbf{0}_{6\times 1}$, $\ddot{\mathbf{q}} = \mathbf{0}_{6\times 1}$) truncated at the first order yields:

$$\mathbf{h}\left(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}\right) \simeq \mathbf{h}(\mathbf{q}_0, \mathbf{0}, \mathbf{0}) + \left.\frac{\partial \mathbf{h}}{\partial \ddot{\boldsymbol{\zeta}}_f}\right|_{(\mathbf{q}_0,\mathbf{0},\mathbf{0})} \ddot{\boldsymbol{\zeta}}_f + \left.\frac{\partial \mathbf{h}}{\partial \ddot{\mathbf{l}}}\right|_{(\mathbf{q}_0,\mathbf{0},\mathbf{0})} \ddot{\mathbf{l}} + \left.\frac{\partial \mathbf{h}}{\partial \dot{\boldsymbol{\zeta}}_f}\right|_{(\mathbf{q}_0,\mathbf{0},\mathbf{0})} \dot{\boldsymbol{\zeta}}_f + \\ + \left.\frac{\partial \mathbf{h}}{\partial \dot{\mathbf{l}}}\right|_{(\mathbf{q}_0,\mathbf{0},\mathbf{0})} \dot{\mathbf{l}} + \left.\frac{\partial \mathbf{h}}{\partial \boldsymbol{\zeta}_f}\right|_{(\mathbf{q}_0,\mathbf{0},\mathbf{0})} \left(\boldsymbol{\zeta}_f - \boldsymbol{\zeta}_{f0}\right) + \left.\frac{\partial \mathbf{h}}{\partial \mathbf{l}}\right|_{(\mathbf{q}_0,\mathbf{0},\mathbf{0})} (\mathbf{l} - \mathbf{l}_0) = \mathbf{0}_{\lambda\times 1} \quad (4.1)$$

At equilibrium, clearly $\mathbf{h}\left(\mathbf{q}_0, \mathbf{0}, \mathbf{0}\right) = \mathbf{f}^{\perp} = \mathbf{0}_{\lambda\times 1}$. The partial derivatives are readily obtained as:

$$\left.\frac{\partial \mathbf{h}}{\partial \ddot{\boldsymbol{\zeta}}_f}\right|_{(\mathbf{q}_0,\mathbf{0},\mathbf{0})} = \mathbf{M}_f^{\perp}\Big|_{(\mathbf{q}_0,\mathbf{0},\mathbf{0})} = \mathbf{M}_{f0}^{\perp} \quad (4.2)$$

$$\left.\frac{\partial \mathbf{h}}{\partial \ddot{\mathbf{l}}}\right|_{(\mathbf{q}_0,\mathbf{0},\mathbf{0})} = \mathbf{M}_l^{\perp}\big|_{(\mathbf{q}_0,\mathbf{0},\mathbf{0})} = \mathbf{M}_{l0}^{\perp} \quad (4.3)$$

$$\left.\frac{\partial \mathbf{h}}{\partial \dot{\boldsymbol{\zeta}}_f}\right|_{(\mathbf{q}_0,\mathbf{0},\mathbf{0})} = \left.\left(\mathbf{C}_f^{\perp} + \frac{\partial \mathbf{C}_f^{\perp}}{\partial \dot{\boldsymbol{\zeta}}_f}\dot{\boldsymbol{\zeta}}_f + \frac{\partial \mathbf{C}_l^{\perp}}{\partial \dot{\boldsymbol{\zeta}}_f}\dot{\mathbf{l}}\right)\right|_{(\mathbf{q}_0,\mathbf{0},\mathbf{0})} = \mathbf{0}_{\lambda\times\lambda} \quad (4.4)$$

$$\left.\frac{\partial \mathbf{h}}{\partial \dot{\mathbf{l}}}\right|_{(\mathbf{q}_0,\mathbf{0},\mathbf{0})} = \left.\left(\mathbf{C}_l^{\perp} + \frac{\partial \mathbf{C}_f^{\perp}}{\partial \dot{\mathbf{l}}}\dot{\boldsymbol{\zeta}}_f + \frac{\partial \mathbf{C}_l^{\perp}}{\partial \dot{\mathbf{l}}}\dot{\mathbf{l}}\right)\right|_{(\mathbf{q}_0,\mathbf{0},\mathbf{0})} = \mathbf{0}_{\lambda\times\lambda} \quad (4.5)$$

$$\left.\frac{\partial \mathbf{h}}{\partial \boldsymbol{\zeta}_f}\right|_{(\mathbf{q}_0,\mathbf{0},\mathbf{0})} = \left.\left(\frac{\partial \mathbf{M}_f^{\perp}}{\partial \boldsymbol{\zeta}_f}\ddot{\boldsymbol{\zeta}}_f + \frac{\partial \mathbf{M}_l^{\perp}}{\partial \boldsymbol{\zeta}_f}\ddot{\mathbf{l}} + \frac{\partial \mathbf{C}_f^{\perp}}{\partial \boldsymbol{\zeta}_f}\dot{\boldsymbol{\zeta}}_f + \frac{\partial \mathbf{C}_l^{\perp}}{\partial \boldsymbol{\zeta}_f}\dot{\mathbf{l}} + \frac{\partial \mathbf{f}^{\perp}}{\partial \boldsymbol{\zeta}_f}\right)\right|_{(\mathbf{q}_0,\mathbf{0},\mathbf{0})} = \\ = \left.\frac{\partial \mathbf{f}^{\perp}}{\partial \boldsymbol{\zeta}_f}\right|_{(\mathbf{q}_0,\mathbf{0},\mathbf{0})} = \mathbf{K}_f^{\perp}\Big|_{(\mathbf{q}_0,\mathbf{0},\mathbf{0})} = \mathbf{K}_{f0}^{\perp} \quad (4.6)$$

$$\left.\frac{\partial \mathbf{h}}{\partial \mathbf{l}}\right|_{(\mathbf{q}_0,\mathbf{0},\mathbf{0})} = \left.\left(\frac{\partial \mathbf{M}_f^{\perp}}{\partial \mathbf{l}}\ddot{\boldsymbol{\zeta}}_f + \frac{\partial \mathbf{M}_l^{\perp}}{\partial \mathbf{l}}\ddot{\mathbf{l}} + \frac{\partial \mathbf{C}_f^{\perp}}{\partial \mathbf{l}}\dot{\boldsymbol{\zeta}}_f + \frac{\partial \mathbf{C}_l^{\perp}}{\partial \mathbf{l}}\dot{\mathbf{l}} + \frac{\partial \mathbf{f}^{\perp}}{\partial \mathbf{l}}\right)\right|_{(\mathbf{q}_0,\mathbf{0},\mathbf{0})} = \\ = \left.\frac{\partial \mathbf{f}^{\perp}}{\partial \mathbf{l}}\right|_{(\mathbf{q}_0,\mathbf{0},\mathbf{0})} = \mathbf{K}_l^{\perp}\big|_{(\mathbf{q}_0,\mathbf{0},\mathbf{0})} = \mathbf{K}_{l0}^{\perp} \quad (4.7)$$

where we have taken advantages that many elements vanishing in Eqs. from (4.2) to (4.7) are linearly dependent on $\dot{\mathbf{q}}$ and $\ddot{\mathbf{q}}$. All quantities at the far right-hand sides from (4.2) to (4.7) are computed in the equilibrium configuration $(\mathbf{q}_0, \mathbf{0}, \mathbf{0})$ and thus denoted by the subscript 0. Matrices $\mathbf{M}_f^{\perp}$, $\mathbf{M}_l^{\perp}$, $\mathbf{K}_f^{\perp}$ and $\mathbf{K}_l^{\perp}$, given in Eqs. (2.72), (2.106) and (2.107), are reported below for the sake of convenience:

$$\mathbf{M}_f^{\perp} = \boldsymbol{\Xi}_l^{\perp T}\mathbf{M}\boldsymbol{\Xi}_l^{\perp} \in \mathbb{R}^{\lambda\times\lambda} \quad (4.8)$$

$$\mathbf{M}_l^{\perp} = \boldsymbol{\Xi}_l^{\perp T}\mathbf{M}\boldsymbol{\Xi}_l^{\parallel} \in \mathbb{R}^{\lambda\times n} \quad (4.9)$$

$$\mathbf{K}_f^{\perp} = \boldsymbol{\Xi}_l^{\perp^T}\left[(\mathbf{K}+\mathbf{Q})\,\mathbf{D} - \mathbf{F}\right]\mathbf{J}_l^{\perp} \quad (4.10)$$

$$\mathbf{K}_l^{\perp} = \boldsymbol{\Xi}_l^{\perp^T}\left[(\mathbf{K}+\mathbf{Q})\,\mathbf{D} - \mathbf{F}\right]\mathbf{J}_l^{\parallel} \quad (4.11)$$

Finally, Eq. (4.1) can be rewritten as:

$$\mathbf{M}_{f0}^{\perp}\ddot{\boldsymbol{\zeta}}_f + \mathbf{M}_{l0}^{\perp}\ddot{\mathbf{l}} + \mathbf{K}_{f0}^{\perp}\left(\boldsymbol{\zeta}_f - \boldsymbol{\zeta}_{f0}\right) + \mathbf{K}_{l0}^{\perp}\left(\mathbf{l} - \mathbf{l}_0\right) = \\ = \mathbf{M}_{f0}^{\perp}\Delta\ddot{\boldsymbol{\zeta}}_{f0} + \mathbf{M}_{l0}^{\perp}\Delta\ddot{\mathbf{l}}_0 + \mathbf{K}_{f0}^{\perp}\Delta\boldsymbol{\zeta}_{f0} + \mathbf{K}_{l0}^{\perp}\Delta\mathbf{l}_0 = \mathbf{0}_{\lambda\times 1} \quad (4.12)$$

where $\Delta\ddot{\boldsymbol{\zeta}}_{f0} = \ddot{\boldsymbol{\zeta}}_f - \mathbf{0}_{\lambda\times 1}$, $\Delta\ddot{\mathbf{l}}_0 = \ddot{\mathbf{l}} - \mathbf{0}_{n\times 1}$, $\Delta\boldsymbol{\zeta}_{f0} = \boldsymbol{\zeta}_f - \boldsymbol{\zeta}_{f0}$, and $\Delta\mathbf{l}_0 = \mathbf{l} - \mathbf{l}_0$.

4.1.2 Linearized free-motion internal-dynamics and natural oscillation frequencies

In case actuators are locked, that is, $\Delta\mathbf{l}_0 = \Delta\ddot{\mathbf{l}}_0 = \mathbf{0}_{\lambda\times 1}$, Eq. (4.12) simplifies to:

$$\mathbf{M}_{f0}^{\perp}\Delta\ddot{\boldsymbol{\zeta}}_{f0} + \mathbf{K}_{f0}^{\perp}\Delta\boldsymbol{\zeta}_{f0} = \mathbf{0}_{\lambda\times 1} \quad (4.13)$$

This formulation leads to a generalized eigenvalue problem, whose solution allows for the determination of the system natural oscillation frequencies in the equilibrium configuration under investigation. By considering a solution of Eq. (4.13) in the form $\Delta\boldsymbol{\zeta}_{f0}(t) = \boldsymbol{\gamma}e^{\Lambda t}$, with $\Lambda \in \mathbb{C}$, so that:

$$\left(\Lambda^2\mathbf{M}_{f0}^{\perp} + \mathbf{K}_{f0}^{\perp}\right)\boldsymbol{\gamma} = \mathbf{0}_{\lambda\times 1} \quad (4.14)$$

the eigenvalues $\Lambda_1^2, \ldots, \Lambda_\lambda^2$ are found by solving the characteristic equation associated to the generalized eigenvalue problem:

$$\det\left(\Lambda^2\mathbf{M}_{f0}^{\perp} + \mathbf{K}_{f0}^{\perp}\right) = 0 \quad (4.15)$$

It should be noted that if and only if $\Lambda_1^2, \ldots, \Lambda_\lambda^2$ are real negative numbers, the equilibrium configuration is stable, otherwise it is unstable [54]: this formulation allows to study the equilibrium stability of non-conservative system.

Finally, natural oscillation frequencies are computed as:

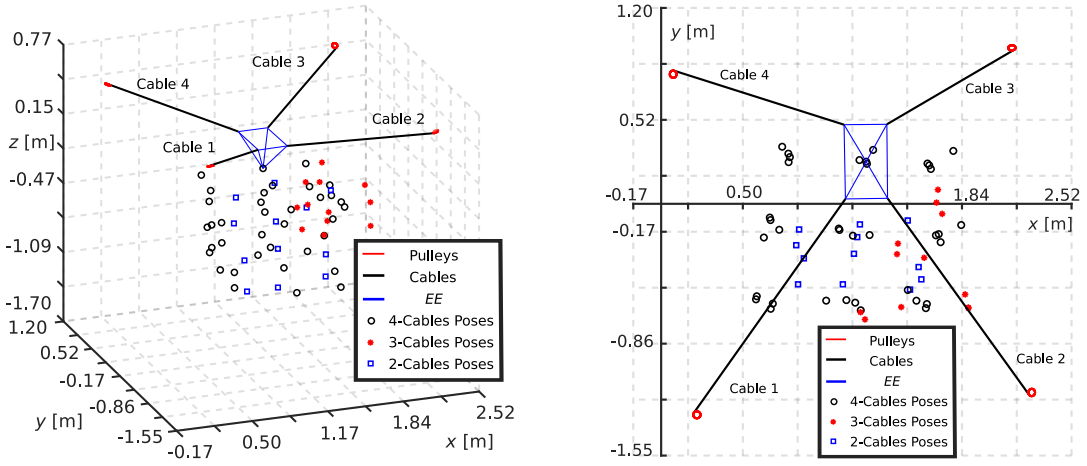
$$f_j = \frac{\Im\left(\Lambda_j\right)}{2\pi}, \qquad j = 1, \ldots, \lambda \quad (4.16)$$

where $\Im(\cdot)$ denotes the imaginary part of a complex number. Additionally, eigenvectors $\boldsymbol{\gamma}_j$ can be determined by solving Eq. (4.14) for any j and normalized according to $\boldsymbol{\gamma}_j^T\mathbf{M}_{f0}^{\perp}\boldsymbol{\gamma}_j = 1$.

4.1.3 Experimental validation

In order to validate the methodology proposed in this section, a series of experiments were conducted on the 6-*DoF UACDPR* prototype of the University of Bologna. Geometrical and inertial properties of the prototype, defined in Sections 2.1 and 2.3, are summarized in Tables B.1 and B.2 in App. B.1. The only external load applied to the robot *EE* is gravity, thus $\mathbf{e}' = \mathbf{s}'$, $\boldsymbol{\phi} = -mg\mathbf{k}$ and $\boldsymbol{\mu} = \mathbf{0}_{3\times 1}$.

The procedure described in this subsection was applied to, respectively: 36 equilibrium configurations in which the platform was constrained by 4 cables (cables 1 to 4), 12 configurations in which only cables 1 through 3 were attached to the platform,



(a) Isometric view (b) Top view

Figure 4.1: Layout of experimental configurations

and 12 configurations in which the platform was suspended only by cables 1 and 3 (Fig. 4.1).

In order to reach an equilibrium configuration, the platform was first displaced in a desired location inside the robot wrench-feasible reachable workspace. Then, actuators were controlled to hold their angular positions so that cable lengths could not change: motor torques were checked to ensure that their values were compatible with cables being taut. The *EE* was then manually slightly displaced with respect to its equilibrium configuration, and swiftly released next: this operation was equivalent to impose non-equilibrium initial conditions to the free-motion dynamics of the platform. The positions $\mathbf{p}_k$, $k = 1,\ldots,5$, of 5 optical markers mounted on the robot platform were tracked by 8 cameras of a VICON Motion Capture System (measurement accuracy was ± 0.2 mm for each marker's Cartesian component, at a 100 Hz sampling rate) for a total duration of 10 s for each experiment, thus acquiring $n_s = 1001$ samples per marker coordinate.

These coordinates were then filtered by using a zero-phase finite-impulse response low-pass digital filter with a stop-band frequency of 10 Hz. No natural oscillation frequency above 4 Hz was expected from the model, thus measurement noise and unmodelled oscillatory phenomena at higher frequencies, such as cable elastic axial vibrations, were accordingly removed.

The n_s *EE* poses corresponding to each experiment were reconstructed from the position of the 5 markers, and the corresponding cable lengths were calculated by the inverse geometric model (see Eq. (2.14)). The mean value over the n_s samples of each cable length differed from its maximum and minimum value by less than 1 mm and thus it was considered as the constant experimental value of the variable. Alternatively, cable lengths could be computed as the result of the inverse model applied to the rest pose of the *EE* that is eventually reached. On the other hand, the procedure employed is considered to be more robust, because static friction may lead the *EE* to stop in an unpredictable configuration.

The natural-oscillation-frequency computation method is summarized as follows:

- given the value $\mathbf{l}_0$ of cable lengths, compute the *EE* static equilibrium pose $\boldsymbol{\zeta}_0$ according to the direct geometric-static model in Eq.(2.86) ($\boldsymbol{\epsilon}$ is expressed by xyz

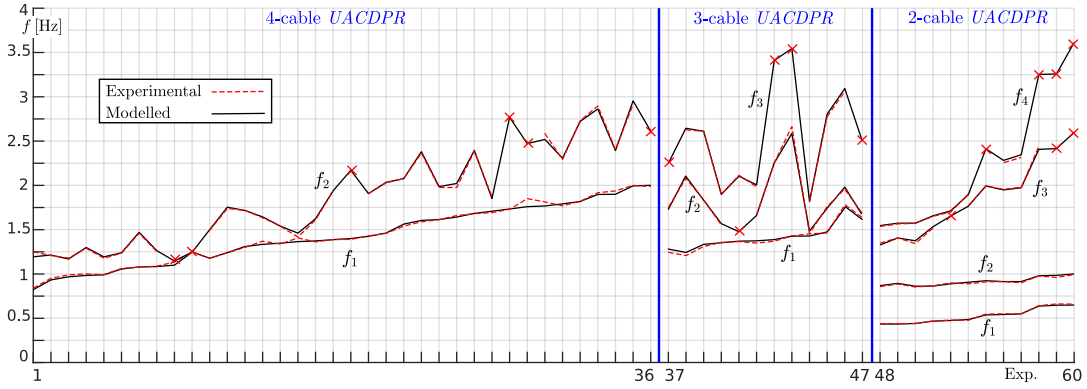


Figure 4.2: Modelled and experimental oscillation frequencies for *UACDPRs* with 2, 3 and 4 cables.

Tait-Bryan angles in this Section); since the problem has possibly multiple solutions, keep only the stable pose that is closer to the initial one;

- once the equilibrium configuration $(\boldsymbol{\zeta}_0, \mathbf{l}_0)$ is known, compute $\mathbf{M}_0^{\perp}$ and $\mathbf{K}_0^{\perp}$ according to Eqs. (4.8) and (4.10);
- solve the generalized eigenvalue problem in Eq. (4.14) and compute the natural oscillation frequencies according to Eq. (4.16).

The oscillation of each marker with respect to its equilibrium position can be experimentally computed as:

$$\Delta\mathbf{p}_k(t) = \mathbf{p}_k(t) - \overline{\mathbf{p}}_k \tag{4.17}$$

where $\overline{(\cdot)}$ denotes the mean value operator. The signal of any coordinate of $\Delta\mathbf{p}_k(t)$ contains, in general, the system natural frequencies since, if $\mathbf{p}_k$ is chosen as the platform reference point, $\Delta\mathbf{p}_k(t)$ can be modelled as:

$$\Delta\mathbf{p}_k(t) = \mathbf{J}_{l,k}^{\perp}\Delta\boldsymbol{\zeta}_{f0}(t) = \mathbf{J}_{l,k}^{\perp}\boldsymbol{\gamma}e^{\Lambda t} = \boldsymbol{\gamma}_k e^{\Lambda t}, \qquad \boldsymbol{\gamma}_k = \mathbf{J}_{l,k}^{\perp}\boldsymbol{\gamma} \tag{4.18}$$

where $\mathbf{J}_{l,k}^{\perp}$ groups the first 3 rows of $\mathbf{J}_l^{\perp}$ as in:

$$\mathbf{J}_l^{\perp} = \mathbf{P}^T\mathbf{J}_P^{\perp} \tag{4.19}$$

and $\mathbf{J}_P^{\perp}$ is given in the left-hand side of Eq. (2.51). Then, the *Fast Fourier Transformation* (*FFT*) of each coordinate of $\Delta\mathbf{p}_k(t)$, for $k = 1,\ldots,5$, can be performed. This operation is deemed necessary since: (i) depending to the actual value of $\boldsymbol{\gamma}_k$, some modes may be absent in some coordinate, (ii) depending on the manually imposed initial condition of the *EE* oscillation, some modes may have an experimentally negligible amplitude in the frequency spectra of a certain coordinate *FFT*, and (iii) high data redundancy, which is achieved by considering 15 signals theoretically possessing frequency spectra peaks corresponding to the same frequency values, robustifies experimental investigation. In the following subsections, several small-amplitude peaks can be noticed surrounding high amplitude-peaks: they are not present in the original signals, but artificially introduced because of an *FFT* resolution upscaling process. In fact, $n_s = 1001$ samples recorded at 100 Hz would produce an *FFT* with 0.1 Hz frequency resolution. This resolution was upscaled to 0.01 Hz in order to better isolate nearby peaks of the signal *FFT*s. This operation was performed by adding, at the end of the n_s recorded samples, $9n_s$ additional zero-value samples, for a total of $10n_s = 10001$ samples.

The results of all experiments are summarized in Fig. 4.2. For each experiment, the experimental natural frequency $f_j^\star$ of the j-th mode ($j = 1,\dots,\lambda$) is determined as the weighted mean of the frequencies $f_{j,kc}^\star$ corresponding to *FFT* peaks of a single coordinate, with the oscillation amplitude A_{kc} used as weight ($k = 1,\dots,5$, $c = x,y,z$):

$$f_j^\star = \left(\sum_{k=1}^{5}\sum_{c=x,y,z} A_{kc} f_{j,kc}^\star\right) \Bigg/ \left(\sum_{k=1}^{5}\sum_{c=x,y,z} A_{kc}\right) \tag{4.20}$$

Accordingly, the standard deviation of the weighted mean is calculated as:

$$\sigma_j^\star = \sqrt{\left(\sum_{k=1}^{5}\sum_{c=x,y,z} A_{kc}(f_{j,kc}^\star - f_j^\star)^2\right) \Bigg/ \left(\sum_{k=1}^{5}\sum_{c=x,y,z} A_{kc}\right)} \tag{4.21}$$

The relative estimation error between the experimental frequency $f_j^\star$ and the modelled one f_j is calculated as:

$$\Delta f_j\% = 100\frac{f_j^\star - f_j}{f_j} \tag{4.22}$$

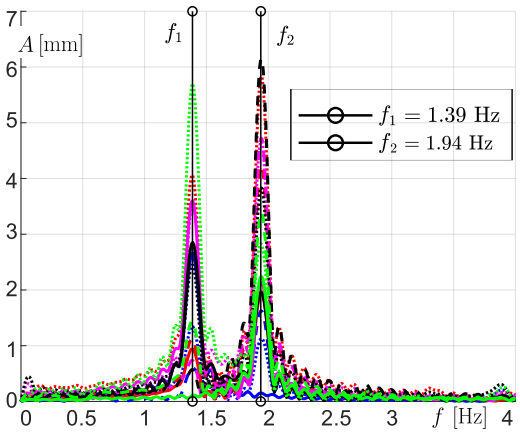
In Fig. 4.2, each integer between 1 and 60 on the abscissa axis represents one of the experimental configurations portrayed in Fig. 4.1, with the ordinate representing the corresponding values of f_j (in black) and $f_j^\star$ (in dashed-red), for $j = 1,\dots,\lambda$. In case no frequency is experimentally detected, a red cross is superimposed to the corresponding modelled frequency. The *FFT*s of two experiments for each architecture are provided in Fig. 4.3 and they will be commented in detail in the following Subsections. For each experiment, the 15 curves corresponding to the x, y, and z coordinate of the 5 markers are superimposed, so as to show the presence of roughly the same experimental natural frequencies in each signal; in addition, the modelled natural frequencies $f_1,\dots,f_\lambda$ are reported in a box and denoted by a black vertical line. Complete experiment data can be found in App. C.1.

4.1.3.1 Results for the 4-Cable *UACDPR*

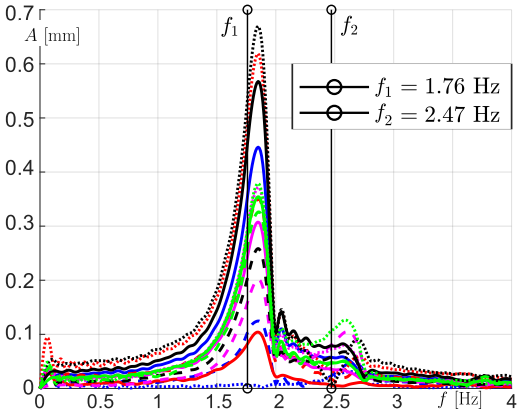
Experimental results for the 4-Cable *UACDPR* are shown in the left side of Fig. 4.2, namely in Experiments 1 through 36. Results are satisfactory, with the largest relative estimation error being 5.15% in Exp. 29 and very low standard deviations, with a maximum value of 0.08 Hz over a 2.91 Hz natural frequency (2.75%) in Exp. 35. In 6 experiments, one out of 2 frequency was not retrievable by data:

- Exp. 9 and 11 have practically coincident natural frequencies for each mode, thus only one peak is distinguishable in their *FFT*s;
- in Exp. 19, 28, 29 and 36, the highest mode was not experimentally excited, thus only one peak is present in their *FFT*s.

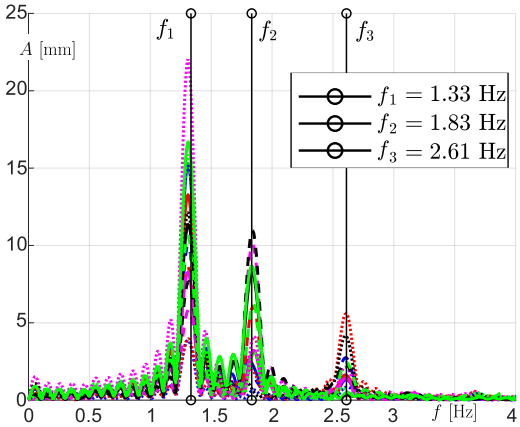
Figure 4.3a shows the best overall result for the 4-cable prototype, obtained in Exp. 18. Experimental frequencies are remarkably close to the modelled one, with a relative estimation error of 0.11% and −0.09%. In addition, standard deviations on experimental frequencies are very small, with less than 0.005 Hz over 1.39 Hz (less than 0.4%) and 0.02 Hz over 1.94 Hz (1%). Figure 4.3b shows the worst overall result for the 4-cable prototype, obtained in Exp. 29. Only one experimental frequency, the lowest, has clearly



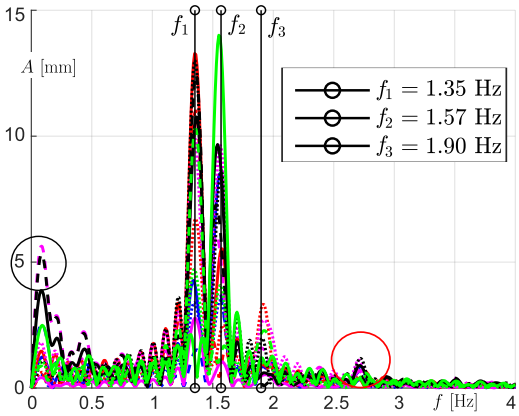
(a) Best overall result on 4-cable *UACDPR*: Experiment 18



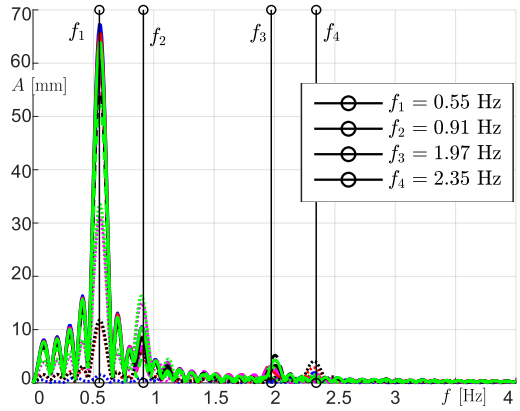
(b) Worst overall result on 4-cable *UACDPR*: Experiment 29



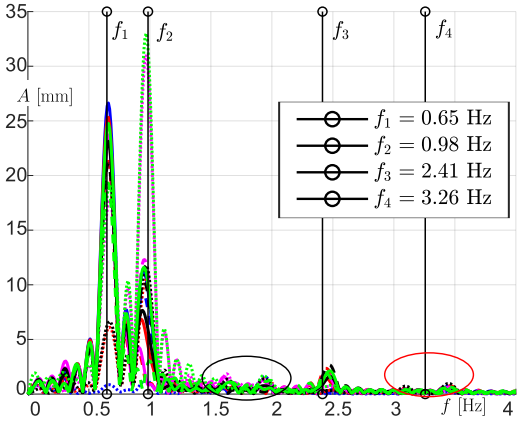
(c) Best overall result on 3-cable *UACDPR*: Experiment 39



(d) Experiment 40 for the 3-cable *UACDPR*: additional *FFT* peaks



(e) Best results on 2-cable *UACDPR*: Experiment 57



(f) Experiment 58 for the 2-cable *UACDPR*: additional *FFT* peaks

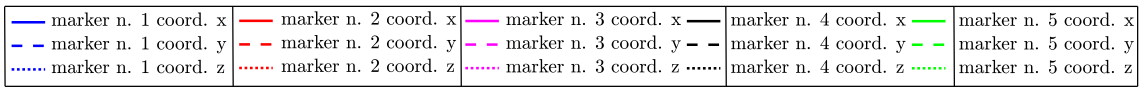


Figure 4.3: Experimental results *FFT*s

noticeable *FFT* peaks, and thus could be reconstructed, with a 0.01 Hz standard deviation over 1.85 Hz (0.5%). On the other hand, the relative estimation error with respect

to the modelled value is 5.11%. These results can be attributed to a very limited oscillation amplitude, which is easily corrupted by unmodelled frictional effects. It was impossible to achieve a larger oscillation amplitude because the configuration was at the edge of the manipulator workspace.

4.1.3.2 Results for the 3-Cable *UACDPR*

Experimental results for the 3-Cable *UACDPR* are shown in the central part of Fig. 4.2, namely in Experiments 37 through 47. Results are still very good, with a maximum relative estimation error of 3% in Exp. 43 and low standard deviations, with a maximum value of 0.07 Hz over a 3.09 Hz natural frequency (2.3%). In 5 experiments, one out of 3 frequencies was not retrievable by data:

- Exp. 44 and 47 have practically coincident natural frequencies for the first two modes, thus only one peak is distinguishable in their *FFT*s;
- in Exp. 39, 44, and 45, the highest mode was not experimentally excited, thus only two peaks are present in their *FFT*s.

Figure 4.3c shows Exp. 39 results, which are the best over-all for the 3-cable prototype. Experimental frequencies are remarkably close to the modelled one, with a relative estimation error of -1.76%, 0.24%, and -0.27%. In addition, standard deviations on experimental frequencies are limited, with less than 0.005 Hz over 1.31 Hz (less than 0.4%), 0.02 Hz over 1.83 Hz (1.1%), and 0.05 Hz over 2.6 Hz (1.9%). Figure 4.3d shows Exp. 40 results, which presents even better results than Exp. 39 in terms of experimental frequency reconstruction, but also shows additional *FFT* peaks, one at roughly 0.08 Hz (circled in black) and another at roughly 2.73 Hz (circled in red). These unexpected peaks can be probably attributed to one cable losing tension during experiments, thus transitioning to a 2-cable architecture. In fact, this may explain both additional *FFT* peaks. On the one hand, an additional peak at a higher frequency may be an additional natural oscillation frequency gained by the loss of a constraint; on the other, the presence of a nearly zero frequency peak is compatible with a non-oscillatory motion from one equilibrium configuration (the one of the 2-cable robot) to another (the one of the 3-cable robot) It should be noted that Exp. 40 was also conducted on the edge of the 3-cable robot workspace. Unfortunately, due to limitations in the experimental equipment, our assumption regarding the loss of tension in one cable cannot be strictly proved.

4.1.3.3 Results for the 2-Cable *UACDPR*

Experimental results for the 2-Cable *UACDPR* are shown in the right side of Fig. 4.2, namely in Experiments 48 through 60. Results are still satisfactory, with a maximum relative estimation error of -2.46% in Exp. 58 and low standard deviations in frequency determination, with a maximum value of 0.07 Hz over a 2.26 Hz natural frequency, (3.1%). In 5 experiments, up to 2 out of 4 frequencies were not retrievable by data:

- Exp. 52 has practically coincident natural frequencies for the last two modes, thus only one peak is distinguishable in their *FFT*s;
- in Exp. 54 and 58 the highest mode was not experimentally excited, thus only three peaks are present in their *FFT*s;

- in Exp. 59 and 60 the two highest mode were not experimentally excited, thus only two peaks are present in their *FFT*s.

Figure 4.3e shows Exp. 57 results, which are the best overall for the 2-cable prototype. Experimental frequencies are remarkably close to the modelled one, with a relative estimation error of 0.05%, −1.46%, 0.33%, and −1.31%. In addition, standard deviations on experimental frequencies are limited, with less than 0.005 Hz over 0.55 Hz (less than 0.9%) and 0.9 Hz (less than 0.6%), 0.03 Hz over 1.98 Hz (1.5%), and 0.06 Hz over 2.31 Hz (2.3%). Figure 4.3f shows Exp. 58 results, with only 3 modes detected, and additional *FFT* peaks of limited amplitude, at roughly 1.6 Hz and 1.9 Hz (circled in black). These small peaks cannot be attributed to the frequency resolution upscaling process, but they are probably related to other low-frequency unmodelled effects, such as cables oscillating out of the pulley planes. Additional oscillatory phenomena are not addressed in this paper and will be the object of future research. On the other hand, they have negligible amplitude compared to the dominant first two modes at lower frequency. Two other small peaks are noticeable around the 4th mode and circled in red.

4.1.3.4 Comments

All in all, experimental results show a remarkable adherence to the model, especially considering the prototype nature of the robots used in the experiments (most structural components, except for the winches, are made of 3D-printed plastic). Moreover, all frictional effects and additional oscillatory phenomena, such as cables oscillating outside the pulley plane, were neglected in our model.

4.2 Inertial Parameter Identification

Classical dynamical parameters identification models aim at determining [72]:

- the mass of moving links, the position of their centers of mass and inertia matrices w.r.t. to some body-fixed frames;
- inertia, friction coefficients, and drive current gains for actuators.

In fact, these parameters are needed in any computed-torque-based controller, which are the mostly employed ones in robotic control in case robot speed and accelerations are high. Several strategies have been proposed in the literature to identify robot dynamic parameters, and most of them present some common practical features [73]:

- a dynamic model that is linear in the inertial parameters is formulated, so that parameters can be algebraically isolated;
- an over-determined linear system of equations is realized, by applying the dynamic model to a sufficient number of configurations along some trajectory of the robot;
- inertial parameters are determined by using linear regression techniques.

On the theoretical side, additional considerations often lead to identification best practices:

- identifiability of inertial parameters can be investigated, so that non-identifiable or non-essential ones are eliminated from the dynamic model [73]; in fact, when a parameter is intrinsically not identifiable or non-essential, it has a minimal influence on the dynamic model: by eliminating it, identifiable parameters are estimated with higher precision and the dynamic model is simplified;

- robot trajectories for identification should be optimal with respect to some identifiability criterion, in order to increase the accuracy of the regression analysis;

- after parameters are identified, statistical error analysis tools should be employed to draw conclusions on the experimental results: in case statistical errors are large, experiments should be disregarded and re-performed.

Since most *CDPRs* employ control schemes which need to keep cable tensions under control, it is no surprise that a growing number of studies include inertial parameter identification in their results. Most notably, standard [74, 75] or simplified [76, 77] identification metodologies are applied to redundantly actuated *CDPRs* or single winches [78]. A few studies also exist on identification of *EE* parameters in robotic systems which are practically coincident to *UACDPRs* [79, 80], even though their target application is inertial-parameter measurements per-se. The authors of [79, 80] developed an instrumentation for the precise measurements of inertial parameters of a suspended body; the instrumentation employs 3 or 4 fixed-length cables, attached to the fixed frame and the mobile platform by universal joints. Thanks to this set-up, encoders on the universal joints connecting the cables to the frame allow for the measurement of cables angular position and ultimately for the estimation of the *EE* pose. Additionally, an axial force sensor, embedded between each universal joint and each cable, measures the cable tension: this measurement is fundamental for the employment of state-of-the-art identification methods. One of the most notable innovations of [79, 80] is the use of free-motion as excitation trajectory for identification.

The need to identify dynamic parameters in an *UACDPR* is justified even if a feedforward control scheme of cable lengths is used. In fact, the computation of actuator set-points relies on the evaluation of the evolution of robot free coordinates while following a trajectory of the controlled ones [45, 46]: the free coordinates are found by numerically integrating the *EE* internal dynamics, which is influenced by the dynamic parameters of the *EE*.

In this Section, the dynamic model linearity in the inertial parameter will be outlined. More specifically, it will be shown that:

1. classical identification models may present difficulties in their application to *CDPRs*, because of the specific design of the actuators of this class of manipulators;

2. the internal dynamics of the *EE* can be used to derive a novel formulation of the identification problem, which is particularly suitable for *UACDPRs*; the internal dynamics does not depend on the *EE* mass, but only on the *EE* center-of-mass location and its specific inertia matrix (that is, its inertia matrix divided by the *EE* mass), and it is linear in these parameters;

3. identification of *EE* center-of-mass location and its specific inertia matrix can be performed without recurring to any force or torque measurement and only requires to directly measure or estimate λ *EE* coordinates, the free ones.

Dynamic parameters will be calculated as the Total Least Square solution of an over-determined system of equations generated from the application of the internal dynamics over a sampled free-motion trajectory. Moreover, optimal free-motion trajectory generation will be investigated and, finally, experimental results on a 4-cable prototype will be discussed.

4.2.1 Internal-dynamics and identification model

The identification of *EE* dynamical parameters starts from the rearrangement of *EE* dynamic equations, so that a linear equation in these parameters is formulated. To this end, we restrict our analysis to an *EE* which is acted upon by gravity only ($\boldsymbol{\phi} = m\mathbf{g}$, $\mathbf{e} = \mathbf{s}$, $\boldsymbol{\mu} = \mathbf{0}_{3\times 1}$), thus Eq. (2.63) may be rewritten as:

$$\begin{bmatrix} m(\ddot{\mathbf{p}} - \mathbf{g}) - m\tilde{\mathbf{s}}'\boldsymbol{\alpha} - m\tilde{\boldsymbol{\omega}}\tilde{\mathbf{s}}'\boldsymbol{\omega} \\ \mathbf{I}_P\boldsymbol{\alpha} + \tilde{\boldsymbol{\omega}}\mathbf{I}_P\boldsymbol{\omega} + m\tilde{\mathbf{s}}'(\ddot{\mathbf{p}} - \mathbf{g}) \end{bmatrix} = -\Xi_l\boldsymbol{\tau} \tag{4.23}$$

The left-hand side of Eq. (4.23) can be algebraically manipulated in order to isolate the *EE* dynamic parameters. First, the vector product rule $\mathbf{a}\times\mathbf{b} = -\mathbf{b}\times\mathbf{a}$ is applied, so that:

$$\begin{bmatrix} m(\ddot{\mathbf{p}} - \mathbf{g}) + m\left(\tilde{\boldsymbol{\alpha}} + \tilde{\boldsymbol{\omega}}\tilde{\boldsymbol{\omega}}\right)\mathbf{s}' \\ \mathbf{I}_P\boldsymbol{\alpha} + \tilde{\boldsymbol{\omega}}\mathbf{I}_P\boldsymbol{\omega} - m(\tilde{\ddot{\mathbf{p}}} - \tilde{\mathbf{g}})\mathbf{s}' \end{bmatrix} = -\Xi_l\boldsymbol{\tau} \tag{4.24}$$

Then, we consider that $\mathbf{s}' = \mathbf{R}\,{}^P\mathbf{s}'$, and $\mathbf{I}_P = \mathbf{R}\,{}^P\mathbf{I}_P\mathbf{R}^T$, where ${}^P\mathbf{s}'$ and ${}^P\mathbf{I}_P$ are constant:

$$\begin{bmatrix} m(\ddot{\mathbf{p}} - \mathbf{g}) + m\left(\tilde{\boldsymbol{\alpha}} + \tilde{\boldsymbol{\omega}}\tilde{\boldsymbol{\omega}}\right)\mathbf{R}\,{}^P\mathbf{s}' \\ \mathbf{R}\,{}^P\mathbf{I}_P\mathbf{R}^T\boldsymbol{\alpha} + \tilde{\boldsymbol{\omega}}\mathbf{R}\,{}^P\mathbf{I}_P\mathbf{R}^T\boldsymbol{\omega} - m(\tilde{\ddot{\mathbf{p}}} - \tilde{\mathbf{g}})\mathbf{R}\,{}^P\mathbf{s}' \end{bmatrix} = -\Xi_l\boldsymbol{\tau} \tag{4.25}$$

If we consider the following identity in the product of a generic symmetric matrix $\mathbf{S}$ and a vector $\mathbf{v}$:

$$\mathbf{S}\mathbf{v} = \bar{\mathbf{v}}\check{\mathbf{S}} \tag{4.26}$$

where:

$$\mathbf{S} = \begin{bmatrix} S_{xx} & S_{xy} & S_{xz} \\ S_{xy} & S_{yy} & S_{yz} \\ S_{xz} & S_{yz} & S_{zz} \end{bmatrix} \qquad \mathbf{v} = \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix} \tag{4.27}$$

$$\bar{\mathbf{v}} = \begin{bmatrix} v_x & 0 & 0 & v_y & v_z & 0 \\ 0 & v_y & 0 & v_x & 0 & v_z \\ 0 & 0 & v_z & 0 & v_x & v_y \end{bmatrix} \qquad \check{\mathbf{S}} = \begin{bmatrix} S_{xx} & S_{yy} & S_{zz} & S_{xy} & S_{xz} & S_{yz} \end{bmatrix}^T \tag{4.28}$$

equation (4.25) can be rewritten as:

$$\begin{bmatrix} m(\ddot{\mathbf{p}} - \mathbf{g}) + m\left(\tilde{\boldsymbol{\alpha}} + \tilde{\boldsymbol{\omega}}\tilde{\boldsymbol{\omega}}\right)\mathbf{R}\,{}^P\mathbf{s}' \\ (\mathbf{R}\overline{{}^P\boldsymbol{\alpha}} + \tilde{\boldsymbol{\omega}}\mathbf{R}\overline{{}^P\boldsymbol{\omega}})\,{}^P\check{\mathbf{I}}_P - m(\tilde{\ddot{\mathbf{p}}} - \tilde{\mathbf{g}})\mathbf{R}\,{}^P\mathbf{s}' \end{bmatrix} = -\Xi_l\boldsymbol{\tau} \tag{4.29}$$

where ${}^P\boldsymbol{\alpha} = \mathbf{R}^T\boldsymbol{\alpha}$ and ${}^P\boldsymbol{\omega} = \mathbf{R}^T\boldsymbol{\omega}$ are the angular acceleration and velocity vectors expressed in the moving frame.

Finally, the *EE* dynamics can be expressed as a linear equation in the *EE* dynamic parameters m, $m\,{}^P\mathbf{s}\prime$, and ${}^P\check{\mathbf{I}}_P$ as:

$$\mathbf{W}_{EE}(\boldsymbol{\zeta}, \dot{\boldsymbol{\zeta}}, \ddot{\boldsymbol{\zeta}})\mathbf{X}_{EE} = -\Xi_l(\boldsymbol{\zeta})\boldsymbol{\tau} \tag{4.30}$$

where:

$$\mathbf{W}_{EE} = \begin{bmatrix} (\ddot{\mathbf{p}} - \mathbf{g}) & (\tilde{\boldsymbol{\alpha}} + \tilde{\boldsymbol{\omega}}\tilde{\boldsymbol{\omega}})\,\mathbf{R} & \mathbf{0}_{3\times 6s} \\ \mathbf{0}_{3\times 1} & -(\tilde{\ddot{\mathbf{p}}} - \tilde{\mathbf{g}})\mathbf{R} & (\mathbf{R}\overline{{}^{P}\boldsymbol{\alpha}} + \tilde{\boldsymbol{\omega}}\mathbf{R}\overline{{}^{P}\boldsymbol{\omega}}) \end{bmatrix}, \qquad \mathbf{X}_{EE} = \begin{bmatrix} m \\ m\,{}^{P}\mathbf{s}\prime \\ {}^{P}\check{\mathbf{I}}_{P} \end{bmatrix} \tag{4.31}$$

If the *EE* motion and cable tensions can be measured or estimated, for example by means of an external photogrammetry system such as the one described in Sec. 4.1.3 or direct kinematics, and force sensors embedded in the cable transmission, the application of Eq. (4.30) over a trajectory which is sampled in n_s configurations leads to the definition of an over-determined system of $6n_s$ equations in 10 unknowns, which can be solved for $\mathbf{X}_{EE}$ and analyzed with the tools of linear regression. It should be noted that, in general, the *EE* pose and its derivatives cannot be inferred by means of direct kinematics only, and additional sensors must be employed if a pose measuring device is to be avoided. As an example, encoders on swivel axes could be employed [44], but their efficacy for pose reconstruction has not been tested in dynamical applications.

On the other hand, in case force sensors are employed, cable tensions are commonly linearly proportional to force sensor currents $\iota_{\tau,i}$, namely:

$$\tau_i = \gamma_{\tau,i}\iota_{\tau,i} \tag{4.32}$$

where $\gamma_{\tau,i}$ is a (possibly) unknown sensor gain. Thus, *EE* parameters and force sensor gains may be determined according to the extended identification model:

$$\mathbf{W}'_{EE}\mathbf{X}'_{EE} = \mathbf{0}_{6\times 1} \tag{4.33}$$

where:

$$\mathbf{W}'_{EE} = \begin{bmatrix} \mathbf{W}_{EE} & \boldsymbol{\Xi}_{\tau} \end{bmatrix} \qquad \mathbf{X}'_{EE} = \begin{bmatrix} \mathbf{X}_{EE} \\ \boldsymbol{\gamma}_{\tau} \end{bmatrix} \tag{4.34}$$

and:

$$\boldsymbol{\gamma}_{\tau} = \begin{bmatrix} \gamma_{\tau,1} & \dots & \gamma_{\tau,n} \end{bmatrix}^{T}, \qquad \boldsymbol{\Xi}_{\tau} = \begin{bmatrix} \boldsymbol{\xi}_1\iota_{\tau,1} & \dots & \boldsymbol{\xi}_n\iota_{\tau,n} \end{bmatrix} \tag{4.35}$$

Even though identification through Eqs. (4.30) and (4.33) is theoretically feasible, there are two main drawbacks in their use:

- force sensors are not particularly accurate in predicting cable tensions, if a simple model such as the one just described is employed, since their readings are disturbed by several factors, the most important of which is friction in the cable transmission;
- if force sensors are not used for manipulator control, embedding accurate force sensor in the *CDPR* is ultimately costly, because they cannot be removed without altering the *CDPR* mechanical model.

A possible solution to this problem is to avoid force sensors, and model winch dynamics so as to correlate cable tensions and motor torques, which are readily available data in any servo-drive. As any single-*DoF* mechanical transmission, the i-th winch can be modelled by a single differential equation:

$$I_{m,i}\ddot{\theta}_{m,i} + c_{m,i}\dot{\theta}_{m,i} = T_{m,i} - T_{\tau,i} \tag{4.36}$$

where $\theta_{m,i}$ is the motor angle, $I_{m,i}$ and $c_{m,i}$ are the total inertia and viscous friction coefficient of the transmission reduced to the motor, respectively, $T_{m,i}$ is the motor

torque, and $T_{\tau,i}$ is the resistant torque produced by the i-th cable on the motor. Note that viscous friction is a common but rather simplified assumption of the real power dissipation in the mechanical transmission, but it works well in practice at medium to high motor speed. By neglecting the friction on pulley bearing[2] (c_m is indeed produced by several other friction sources in the winch), and recalling that cables are modelled as non-extensible, we may introduce the winch-cable transmission ratio η_i, thus having:

$$T_{\tau,i} = \tau_i / \eta_i \tag{4.37}$$

In addition, in case a standard servo-motor is used to drive winch mechanics, its torque is usually proportional to a drive gain $\gamma_{m,i}$ and to drive electrical current $\iota_{m,i}$:

$$T_{m,i} = \gamma_{m,i} \iota_{m,i} \tag{4.38}$$

Thus, substituting Eqs. (4.37) and (4.38) in Eq. (4.36) one gets:

$$I_{m,i}\ddot{\theta}_{m,i} + c_{m,i}\dot{\theta}_{m,i} = \gamma_{m,i}\iota_{m,i} - \tau_i/\eta_i \tag{4.39}$$

and, upon multiplying both sides by the constant and non-zero η_i:

$$I_{\eta,i}\ddot{\theta}_{m,i} + c_{\eta,i}\dot{\theta}_{m,i} = \gamma_{\eta,i}\iota_{m,i} - \tau_i \tag{4.40}$$

where:

$$I_{\eta,i} = I_{m,i}\eta_i, \qquad c_{\eta,i} = c_{m,i}\eta_i, \qquad \gamma_{\eta,i} = \gamma_{m,i}\eta_i \tag{4.41}$$

Motor angle $\theta_{m,i}$ and current $\iota_{m,i}$ are available measurements in servo-controlled applications, thus they can be used for identification purposes. Finally, τ_i linearly depend on winch parameters $I_{\eta,i}$, $c_{\eta,i}$, $\gamma_{\eta,i}$:

$$\mathbf{W}_{\eta,i}\mathbf{X}_{\eta,i} = -\tau_i \tag{4.42}$$

where:

$$\mathbf{W}_{\eta,i} = \begin{bmatrix}\ddot{\theta}_{m,i} & \dot{\theta}_{m,i} & -\iota_i\end{bmatrix}, \qquad \mathbf{X}_{\eta,i} = \begin{bmatrix}I_{\eta,i} & c_{\eta,i} & \gamma_{\eta,i}\end{bmatrix}^T \tag{4.43}$$

The complete identification model for each winch is readily obtained by considering the n decoupled winch dynamic equations:

$$\mathbf{W}_\eta \mathbf{X}_\eta = -\boldsymbol{\tau} \tag{4.44}$$

where:

$$\mathbf{W}_\eta = \begin{bmatrix}\mathbf{W}_{\eta,1} & \mathbf{0}_{1\times3} & \dots & \mathbf{0}_{1\times3} \\ \mathbf{0}_{1\times3} & \mathbf{W}_{\eta,2} & \dots & \mathbf{0}_{1\times3} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0}_{1\times3} & \dots & \mathbf{0}_{1\times3} & \mathbf{W}_{\eta,n}\end{bmatrix}, \qquad \mathbf{X}_\eta = \begin{bmatrix}\mathbf{X}_{\eta,1} \\ \mathbf{X}_{\eta,2} \\ \vdots \\ \mathbf{X}_{\eta,n}\end{bmatrix} \tag{4.45}$$

In the end, Eq. (4.44) can be substituted in Eq. (4.30), yielding:

$$\mathbf{W}_{EE}\mathbf{X}_{EE} - \Xi_l \mathbf{W}_\eta \mathbf{X}_\eta = \mathbf{0}_{6\times1} \tag{4.46}$$

namely:

$$\mathbf{W}_m \mathbf{X}_m = \mathbf{0}_{6\times1} \tag{4.47}$$

[2]The inclusion of such effects, which are fairly limited, would make the model non-linear.

where:

$$\mathbf{W}_m = \begin{bmatrix} \mathbf{W}_{EE} & -\Xi_l \mathbf{W}_\eta \end{bmatrix}, \qquad \mathbf{X}_m = \begin{bmatrix} \mathbf{X}_{EE}^T & \mathbf{X}_\eta^T \end{bmatrix}^T \tag{4.48}$$

Equation (4.47) is commonly employed for identification purposes [72], since it uses standard embedded sensors, such as encoders on motor axes and drive current measurements, as well as a pose measuring device or algorithm: thanks to these measured quantities (and their numerical differentiation) matrix $\mathbf{W}_m$ may be known at any time. The main drawbacks in the use of Eq. (4.47) for *CDPR* parameter identification are related to winch dynamics:

- common winch design [12, 81] include multiple rototranslating elements, which represent severe sources of friction: experiments in our laboratory showed that this contribution may represent up to 70% of the torque produced by the actuator, even at low speed;
- high friction is a serious problem in identification because of its unpredictable nature; the transition between static and kinematic friction adds high frequency effects to motor current signals, which needs to be carefully filtered; moreover, the linear model outlined in this section is only valid at high motor speed, a condition which effectively limits identification trajectories.

Thus, an alternative formulation of the identification model is hereby proposed, which aims at determining *EE* dynamic parameters *only*, thus avoiding to measure cable tensions (resp. force sensor currents) and motor torques (resp. drive currents). We start by observing that if Eq. (4.30) is pre-multiplied by $\Xi_l^{\perp T}$, the right-hand side of Eq. (4.30) vanishes:

$$\Xi_l^{\perp T} \mathbf{W}_{EE} \mathbf{X}_{EE} = \mathbf{0}_{\lambda \times 1} \tag{4.49}$$

Identification equations are fewer in Eq. (4.49), λ, than in Eq. (4.30), n, thus the identification trajectory should be sampled in more points $n_s' > n_s$, but no force or torque measurement is now required. In addition, since the internal dynamics is linear in $\mathbf{X}_{EE}$ and homogeneous, its validity is not impacted by the multiplication or division by a non-zero scalar. Thus, we can deduce that the *EE* internal dynamics is not influenced by the *EE* mass, and we can divide Eq. (4.49) by m:

$$\mathbf{W}_{EE}' \mathbf{X}_{EE}' = \mathbf{0}_{\lambda \times 1} \tag{4.50}$$

where:

$$\mathbf{W}_{EE}' = \Xi_l^{\perp T} \mathbf{W}_{EE}, \qquad \mathbf{X}_{EE}' = \begin{bmatrix} 1 \\ {}^P\mathbf{s}' \\ {}^P\check{\mathbf{I}}_P' \end{bmatrix}, \qquad {}^P\check{\mathbf{I}}_P' = {}^P\check{\mathbf{I}}_P / m \tag{4.51}$$

It should be noted that in case the *EE* mass is needed, it can be inferred by detaching the *EE* and by weighting it. In case the detachment of the platform is not possible, an alternative solution could be to add a known payload of mass Δm to the *EE* in a known location ($^P\mathbf{p}_L'$ is the position vector of the payload center of mass in the moving frame) and identify the new *EE* with additional payload. If the total center of mass of the *EE* with the additional payload is denoted by $^P\mathbf{s}_L'$, the definition of center of mass gives:

$$(m + \Delta m)\, {}^P\mathbf{s}_L' = m\, {}^P\mathbf{s}' + \Delta m\, {}^P\mathbf{p}_L' \tag{4.52}$$

and the *EE* mass m can be determined from:

$$m({}^{P}\mathbf{s}'_{L} - {}^{P}\mathbf{s}') = \Delta m({}^{P}\mathbf{p}'_{L} - {}^{P}\mathbf{s}'_{L}) \tag{4.53}$$

The knowledge of the standard deviations associated with the identified ${}^{P}\mathbf{s}'_{L}$ and ${}^{P}\mathbf{s}'$ could provide a weighting strategy for the solution of the over-determined system of equations given in Eq. (4.53).

Equation (4.50) is valid in general for *UACDPRs*, but it can be further specialized in order to simplify experimental identification. In fact, *in case the EE exciting trajectory is a free-motion one*, the expression of $\boldsymbol{\zeta}$ and its derivatives, which are needed in order to compute the elements of $\mathbf{W}'_{EE}$, depends only on the *constant* value of the cable lengths, $\mathbf{l}_0$, and on the λ free coordinates of the *EE* $\boldsymbol{\zeta}_f$ (and its derivatives).

4.2.2 Total least square identification

Equation (4.50) is applied to n'_s configurations deriving from one or more free-motion trajectories. Usually, the n'_s samples are obtained by over-sampling *EE* free coordinates at high frequency, then by band-pass filtering them, and finally by decimating them. This results in the over-determined system of equations:

$$\mathbf{W}\mathbf{X} = \mathbf{0}_{\lambda n'_s \times 1}, \qquad \mathbf{W} = \begin{bmatrix} \mathbf{W}'_{EE,1} \\ \vdots \\ \mathbf{W}'_{EE,n'_s} \end{bmatrix} \tag{4.54}$$

where matrix $\mathbf{W} \in \mathbb{R}^{\lambda n'_s \times 10}$ and vector $\mathbf{X} = \mathbf{X}'_{EE} \in \mathbb{R}^{10\times 1}$. There are two primary source of errors which are not considered in the ideal Eq. (4.54): model errors, and also measurement errors. Indeed, Eq. (4.54) has two principal sources of measurement errors, which are the numerical differentiation for velocity and acceleration determination, and the strong dependence on kinematic parameters, since matrix $\boldsymbol{\Xi}_l^{\perp}$ is employed. Model errors are accounted for by considering that the right-hand side of Eq. (4.54) is not a zero vector, but an error vector $\boldsymbol{\varepsilon} \in \mathbb{R}^{\lambda n'_s \times 1}$, while measurement errors affects matrix $\mathbf{W}$, which can be more realistically modelled as $\mathbf{W} + \Delta\mathbf{W}$:

$$(\mathbf{W} + \Delta\mathbf{W})\mathbf{X} = \boldsymbol{\varepsilon} \tag{4.55}$$

The Total Least Square (*TLS*) [82] solution of Eq. (4.55) is therefore considered. This technique allows to compute $\mathbf{X}$ while also minimizing $\Delta\mathbf{W}$ and $\boldsymbol{\varepsilon}$. According to the *TLS* technique, the real system $(\mathbf{W} + \Delta\mathbf{W})\mathbf{X} = \boldsymbol{\varepsilon}$ is changed to its closest compatible system of the form:

$$\hat{\mathbf{W}}\hat{\mathbf{X}} = \mathbf{0}_{\lambda n'_s \times 1} \tag{4.56}$$

where $\text{rank}(\hat{\mathbf{W}}) = 9 < 10 = \text{rank}(\mathbf{W})$, and it is closest to $\mathbf{W}$ with respect to the Frobenius norm, i.e. $\hat{\mathbf{W}}$ minimizes the Frobenius norm $\|\hat{\mathbf{W}} - \mathbf{W}\|_F$. Accordingly, $\hat{\mathbf{X}}$ is the solution of Eq. (4.56) and the *TLS* solution of Eq. (4.55).

$\hat{\mathbf{W}}$ can be straightforwardly computed by performing the *Singular Value Decomposition (SVD)* of $\mathbf{W}$:

$$\mathbf{W} = \mathbf{U}\begin{bmatrix} \mathbf{S} \\ \mathbf{0}_{(\lambda n'_s - 10)\times 10} \end{bmatrix}\mathbf{V}^T \tag{4.57}$$

where $\mathbf{U}$ and $\mathbf{V}$ are, respectively, $(\lambda n'_s \times \lambda n'_s)$ and 10×10 orthonormal matrices, and $\mathbf{S}$ is the (10×10) diagonal matrices of $\mathbf{W}$ singular values (which we assume to be sorted in decreasing order). Then, $\hat{\mathbf{W}}$ is calculated as [82]:

$$\hat{\mathbf{W}} = \mathbf{W} - s_{10}\mathbf{U}_{10}\mathbf{V}_{10}^T \tag{4.58}$$

where s_{10} is the smallest singular value of $\mathbf{W}$, and $\mathbf{U}_{10}$ and $\mathbf{V}_{10}^T$ are the *10*-th column of $\mathbf{U}$ and $\mathbf{V}$, respectively. Then, the *TLS* solution of Eq. (4.54) is given by:

$$\mathbf{X} = \mathbf{V}_{10} \tag{4.59}$$

where $\mathbf{X}$ is normalized so as to have 1 in its first element, according to Eq. (4.50).

Standard deviations σ_{X_i}, with $i = 2, \ldots, 10$, on the dynamic parameters are estimated assuming that errors in the identification matrix $\mathbf{W}$ are independent and identically distributed with zero mean and common covariance σ_W^2 [73]. An unbiased estimator of the standard deviation σ_W is given by [82]:

$$\sigma_W = \frac{s_{10}}{\sqrt{\lambda n'_s - 10}} \tag{4.60}$$

and the covariance matrix of the *TLS* solution error is approximated by:

$$\mathbf{C}_X = \sigma_W^2 \left(1 + \|\mathbf{X}_{2:10}\|_2^2\right) \left(\hat{\mathbf{W}}_{2:10}^T \hat{\mathbf{W}}_{2:10}\right)^{-1} \tag{4.61}$$

with $\mathbf{X}_{2:10}$ contains every element of $\mathbf{X}$ except the first one, and $\hat{\mathbf{W}}_{2:10}$ contains all the columns of $\hat{\mathbf{W}}$ except the first one. Finally, standard deviations on the dynamic parameters are given by:

$$\sigma_{X_i} = \sqrt{\mathbf{C}_X(i,i)}, \qquad i = 2, \ldots, 10 \tag{4.62}$$

where $\mathbf{C}_X(i,i)$ is the i-th diagonal term of $\mathbf{C}_X$, and its relative value with respect to the identified parameter is:

$$\sigma_{\%X_i} = 100\sigma_{X_i} / \|\mathbf{X}_i\| \tag{4.63}$$

If the value of $\sigma_{\%X_i}$ is lower than 5%, the corresponding parameter is commonly considered to be well identified [73]. There are mainly 2 scenarios in which the values of $\sigma_{\%X_i}$ may be large, which require different additional steps:

- in case the parameter $\mathbf{X}_i$ is near zero, $\sigma_{\%X_i}$ could naturally be very large: this kind of parameters are called non-essentials and can be removed from the dynamic model because their influence is very limited; by removing them from vector $\mathbf{X}$ and the corresponding column from matrix $\mathbf{W}$, the *TLS* analysis can be re-performed with possibly higher-accuracy results;

- in case the parameter $\mathbf{X}_i$ is not near zero, it means something went wrong during experiments, or the parameters are non-identifiable: in the former case, experiments should be re-performed, whereas the latter case requires different modelling strategy and experiments.

4.2.3 Optimal free-motion excitation

The problem of generating optimal exciting motion was extensively studied [83], and it commonly aims at *determining robot actuator motion laws* by the *constrained nonlinear optimization of some cost function* correlated with the identification problem [84]. Thus, experiment design focuses on two aspects: how to excite the system, and what to optimize.

Actuator motion laws may be parametric polynomials [83], B-splines [85], sinusoidal [86] or other functions, so that the parameters upon which they depend can be determined as the solution of the optimal excitation problem. Depending on the manipulator under study, the choice of a type of trajectory may have specific advantages. For example, if the joints of a serial manipulator are excited by sinusoidal motion laws, small control errors on joint angles may introduce noise in joint speed and acceleration. Since the nominal trajectory is periodic and band-limited, it is quite easy to design a post-processing filter aiming at removing undesired noise from speed and acceleration signals, which are needed for identification [86]. In our case, we chose to apply the exiting motion not to actuators, but to free-coordinates: the optimal identification results reported in [79] demonstrated this approach to be both feasible and effective for *UACDPRs*. This motion is naturally sinusoidal *if the amplitude is limited*. Indeed, Sec. 4.1.3 analyzed *EE* small-amplitude free motion about equilibrium configurations, and experimentally verified that its sinusoidal approximation (c.f. Eq. (4.13) through Eq. (4.16)) is true in practice. Thus the *EE* small-amplitude free-motion has the same benefits as actuator sinusoidal excitation, i.e. ease of filtering. Thus, *EE* small-amplitude free-motion is chosen as exciting trajectory.

In the following, *EE* small-amplitude free motion is characterized in more details. Each mode j of the free coordinates has the form:

$$\Delta\boldsymbol{\zeta}_{f0,j}(t) = \boldsymbol{\gamma}_j \mathrm{e}^{\Lambda_j t} \tag{4.64}$$

Since only stable equilibrium configurations are chosen as configurations about whom the *EE* oscillates in free motion, the eigenvalues determined by solving Eq. (4.14) are real negative number. This means that:

$$\Lambda_j = \pm\sqrt{-\|\Lambda_j^2\|} = \pm i 2\pi f_j \tag{4.65}$$

with i being the imaginary unit. Since both solutions $\pm i 2\pi f_j$ correspond to the same eigenvector $\boldsymbol{\gamma}_j$, the general response of each mode is:

$$\Delta\boldsymbol{\zeta}_{f0,j}(t) = A_j \boldsymbol{\gamma}_j \cos(2\pi f_j t - \varphi_j) \tag{4.66}$$

with A_j and φ_j being oscillation amplitude and phase, to be determined according to free-motion initial conditions. The total free-motion response of the *EE* is thus:

$$\Delta\boldsymbol{\zeta}_{f0}(t) = \sum_{j=1}^{\lambda} A_j \boldsymbol{\gamma}_j \cos(2\pi f_j t - \varphi_j) \tag{4.67}$$

In the end, we may conclude that the parameters to be optimized for a free-motion excitation are A_j, $\boldsymbol{\gamma}_j$, f_j and φ_j, for $j = 1,\ldots,\lambda$. On the other hand, these parameters cannot be physically selected during experiment, since they depend on other physical quantities, which are to be optimized on their behalf. More specifically, these parameters depend on:

- *UACDPR* cable lengths, whose fixed value allows the equilibrium configuration, and consequently $\boldsymbol{\gamma}_j$ and f_j, to be determined;

- initial displacement and velocity of the free coordinates, which are the initial conditions of the free-motion internal dynamics, and allow A_j and φ_j to be computed.

The total number of parameters to be determined is thus $n+2\lambda$, that is, n cable lengths $\mathbf{l}_0$ and 2λ free-motion initial conditions $\Delta\boldsymbol{\zeta}_{f0}(0)$ and $\Delta\dot{\boldsymbol{\zeta}}_{f0}(0)$. In order to simplify he experiment design, $\Delta\dot{\boldsymbol{\zeta}}_{f0}(0)$ can be chosen to be the zero vector[3], which means to assign a non-zero initial velocity to the *EE* free-motion. Accordingly, the *EE* should be displaced w.r.t. its equilibrium configuration, kept still to make its velocity vanish, and then left to oscillate. In addition, it should be noted that the oscillation about a single equilibrium configuration is unlikely to result in a satisfactory identification, since the resulting identification matrix would be biased: therefore, a better option is to let *EE* oscillate about n_e different equilibria, so that the optimal excitation algorithm should determine $\mathbf{l}_{0,k}$ and $\Delta\boldsymbol{\zeta}_{f0,k}(0)$ ($k=1,\dots,n_e$), for a total of $n_e(n+\lambda)$ parameters. Note that, in case an equal number of samples per equilibria n_s^e, the total number of samples per identification would be $n_s' = n_s^e n_e$.

The cost function to be minimized in order to determine optimal excitation parameters is always correlated to two main factors:

- the elements of the covariance matrix in Eq. (4.61) should be small: the smaller its elements, the higher is the accuracy of the identification [87];

- the numerical solution of the identification problem, that is, the SVD decomposition of $\mathbf{W}$, should be stable: slight changes in the excitation trajectory should not negatively affect the identification problem solution [88].

Practically, both of these issues are tackled by minimizing:

$$C = \frac{s_1}{s_9} + \frac{1}{s_9} = \frac{s_1+1}{s_9} \tag{4.68}$$

Since s_1 and s_9 are the largest and smallest non-zero singular value of $\hat{\mathbf{W}}$, minimizing s_1/s_9 amounts at requiring that the condition number of $\hat{\mathbf{W}}$ is minimum[4]. Since the aforementioned ratio could be minimum for small or large singular values alike, the additional minimization of $1/s_9$ amount at requiring the singular values to be as large as possible. In fact, the magnitude of matrix $\left(\hat{\mathbf{W}}_{2:10}^T\hat{\mathbf{W}}_{2:10}\right)^{-1}$ elements depends on the inverse of the singular values [88]. Thus, we propose the non-linear optimization problem:

$$\left[\mathbf{l}_{0,1},\dots,\mathbf{l}_{0,n_e},\Delta\boldsymbol{\zeta}_{f0,1}(0),\dots,\Delta\boldsymbol{\zeta}_{f0,n_e}(0)\right] = \operatorname{argmin}(C) \tag{4.69}$$

subject to:

$$\begin{cases} \tau_i(t) \le \tau_M, & \forall i,t \\ \tau_i(t) \ge \tau_m, & \forall i,t \\ -\Delta\boldsymbol{\zeta}_{fL} \le \Delta\boldsymbol{\zeta}_{f0}(t) \le \Delta\boldsymbol{\zeta}_{fL}, & \forall t \end{cases} \tag{4.70}$$

[3] This implies $\varphi_j = 0$, for every j.

[4] Note that the minimum value of the condition number is 1 [49]

where τ_i is the i-th cable tension, τ_m and τ_M are minimum and maximum cable tension limits, and $\Delta\boldsymbol{\zeta}_{fL}$ is an upper limit for the oscillation of the free coordinates. These constraints essentially require the trajectory to be (dynamically) wrench-feasible and oscillations to have limited amplitude. The theoretical absolute minimum value of the cost function is 1: this information is useful because it allows us to determine how close to theoretical optimality a real experiment is. A value of $C < 100$ is typically considered good in practice, and $C < 10$ is optimal [83, 89].

A last remark is about computation. The elements of $\mathbf{W}$ require the computation of the *EE* pose: since oscillations are constrained to be small and we verified that the linearized model of Sec. 4.1.3 work well in practice, the *EE* pose can be determined, about any equilibrium configuration, as:

$$\boldsymbol{\zeta}(t) = \boldsymbol{\zeta}_0 + \Delta\boldsymbol{\zeta}(t) \tag{4.71}$$

where $\Delta\boldsymbol{\zeta}(t) = \mathbf{J}_{l,0}^{\perp}\Delta\boldsymbol{\zeta}_{f0}(t)$, $\mathbf{J}_{l,0}^{\perp}$ is calculated in the equilibrium configuration, and $\Delta\boldsymbol{\zeta}_{f0}(t)$ is as in Eq. (4.67). In this way, the numerical integration of the free-motion internal dynamics is avoided and computation is sped up. In addition, the computation of $\tau_i(t)$ can similarly be carried out as:

$$\boldsymbol{\tau}(t) = \boldsymbol{\tau}_0 + \Delta\boldsymbol{\tau}_0(t) \tag{4.72}$$

where $\boldsymbol{\tau}_0$ is the tension vector value at equilibrium, calculated as in Eq. (2.84), and $\Delta\boldsymbol{\tau}_0(t)$ may be computed by linearizing Eq. (2.81) about the equilibrium configuration (see App. A.10 for the derivation):

$$\Delta\boldsymbol{\tau}_0(t) = \mathbf{M}_{f0}^{\parallel}\Delta\ddot{\boldsymbol{\zeta}}_{f0}(t) + \mathbf{K}_{f0}^{\parallel}\Delta\boldsymbol{\zeta}_{f0}(t) \tag{4.73}$$

where the subscript 0 denotes that matrices are constant and evaluated in the equilibrium configuration under consideration and:

$$\mathbf{M}_f^{\parallel} = -{\boldsymbol{\Xi}_l^{\parallel}}^T\mathbf{M}\boldsymbol{\Xi}_l^{\perp}, \qquad \mathbf{K}_f^{\parallel} = -{\boldsymbol{\Xi}_l^{\parallel}}^T\left[(\mathbf{K}+\mathbf{Q})\,\mathbf{D}-\mathbf{F}\right]\mathbf{J}_l^{\perp} \tag{4.74}$$

4.2.4 Optimal excitation computation

The methodology proposed in this section was applied for the determination of the inertial parameters of a 4-cable 6-*DoF UACDPR* prototype of the University of Bologna, whose geometrical properties are defined in Section 2.1 and summarized in Table B.1 in App. B.1. The only external load applied to the robot *EE* was gravity, thus $\mathbf{e}' = \mathbf{s}'$, $\boldsymbol{\phi} = -mg\mathbf{k}$ and $\boldsymbol{\mu} = \mathbf{0}_{3\times1}$.

The x and y coordinates of the *EE* reference point were chosen as *UACDPR* free coordinates. This choice aims at minimizing experimental effort, thus cost and complexity: these coordinates are straightforwardly recorded by an external measurement system, if a marker is placed on the reference point. Orientation measurements would require additional markers mounted on the platform and a mathematical model aiming at extracting orientation information from the relative position of points. Tension and oscillation limits were set to $\tau_m = 20$N, $\tau_M = 200$N and $\Delta\boldsymbol{\zeta}_{fL} = [0.1, 0.1]^T$m. The trajectory optimization problem in Eq. (4.69) was solved considering nominal dynamic

Table 4.1: Optimized cable lengths (in [m])

k	1	2	3	4	5	6	7	8	9	10	11	12
l_1	1.66	1.32	1.30	1.52	1.11	1.38	1.19	1.16	1.46	1.43	1.35	1.52
l_2	1.86	1.58	1.26	1.26	1.68	1.39	1.15	1.17	1.16	1.37	1.45	1.12
l_3	1.68	1.99	1.57	1.27	2.19	1.36	1.65	1.63	1.24	1.27	1.42	1.17
l_4	1.41	1.76	1.56	1.50	1.72	1.32	1.65	1.59	1.50	1.30	1.28	1.54

Table 4.2: Optimized free coordinate displacements $\Delta\boldsymbol{\zeta}_{f0}(0)$ (in [mm])

k	1	2	3	4	5	6	7	8	9	10	11	12
x	5.01	4.34	39.3	76.0	4.22	25.3	1.97	24.9	43.8	47.0	2.37	96.9
y	2.21	7.11	17.6	14.4	7.95	16.6	10.9	9.70	17.4	1.51	11.7	99.7

parameters estimated by *CAD*)

$$ {}^{P}\mathbf{s}\prime = \begin{bmatrix} 0 \\ 0 \\ 0.19 \end{bmatrix} \text{ m}, \qquad {}^{P}\check{\mathbf{I}}'_{P} = \begin{bmatrix} 0.051 \\ 0.069 \\ 0.037 \\ 0 \\ 0 \\ 0 \end{bmatrix} \text{ Kg}\cdot\text{m}^2 \tag{4.75} $$

Since Eq.(4.69) is nonlinear and its gradient is not readily available, it was solved numerically by using *fmincon* MATLAB function, which employs an interior-point algorithm [90] and numerically estimates cost function gradient. Additionally, a *multistart* algorithm was employed in order to automatically provide 100 randomly selected initial guesses for $\mathbf{l}_{0,k}$ and $\Delta\boldsymbol{\zeta}_{f0,k}(0)$, with $k = 1,\ldots,n_e$. Note that the numerical solution of the optimization problem requires also to chose the number of equilibrium configuration, n_e, and the number of samples per equilibrium, n_s^e. They were tuned to $n_e = 12$ and $n_s^e = 100$ for optimal results, as discussed below.

The cost function was optimized to $C = 5.94$, resulting in cable lengths as in Tab. 4.1 and free coordinate displacements as in Tab. 4.2. This results is certainly good enough w.r.t. identification best practices [83, 89].

It should be noted that the multistart algorithm provided a large number of local minima with a cost function value $C < 10$ (the multistart optimization was run several times for robustness, with negligible variations in results), showing that:

- optimization results are very limitedly influenced by the choice of equilibria upon which the *EE* oscillate: every time the optimization was solved, different optimal values for cable lengths and initial free coordinate displacements were found, most of which resulted in a cost function value $C < 10$;
- the different equilibrium configurations about which the platform oscillates are randomly positioned in the robot wrench-feasible reachable workspace: this fact in probably related to a low bias typically induced by randomness;
- minimal value of C are found for large free-coordinate oscillations: their maximum values are such that cable tension values meet their limits for some t.

Therefore, randomly selecting *UACDPR* equilibrium configurations and letting the platform oscillate without cables loosing tension is expected to provide optimal identification results, since the values of parameters resulting in a cost function $C < 10$ are equally optimal in practice.

On the other hand, the number of different equilibria n_e upon which the *EE* should oscillate, and the number of samples per equilibria n_s^e, appear to have far a larger influence on the cost function minimal value. Simulations showed that a number of equilibria between 8 and 16, and a number of samples per equilibria n_s^e between 50 and 150, performed satisfactorily, resulting in $C < 50$. $n_e = 12$ and $n_s^e = 100$ were thus chosen because they provided $C < 10$ in simulations most of the times. This values were not chosen by performing a rigorous analysis, which is left as a future study.

4.2.5 Identification experiments and results

Experiments were performed in order to validate the proposed strategy for identification of *UACDPR* dynamic parameters. Based on the results of the previous Section, experimental analysis was conducted on randomly selected configurations. The platform was displaced in 12 random equilibrium configurations inside the robot wrench-feasible reachable workspace. Then, actuators were controlled to hold their angular positions so that cable lengths could not change: encoder angular positions were recorded so as to determine the experimental value of the cable lengths $\mathbf{l}_0^\star$, and motor torques were checked to ensure that their values were compatible with cables being taut.

The *EE* was then manually slightly displaced with respect to its equilibrium configuration, and released next. The positions of the optical marker placed onto the *EE* reference point was tracked by 8 cameras of a VICON Motion Capture System (measurement accuracy was ±0.2 mm for each marker's Cartesian component, at a 100 Hz sampling rate) for a total duration of 10 s for each experiment, thus acquiring 1001 samples per marker coordinate. Marker position initial value before oscillations were regarded as the equilibrium position of the *EE*.

These recorded coordinates were then filtered by using a zero-phase finite-impulse response low-pass digital filter with a stop-band frequency of 10 Hz. Then, signals were numerically differentiated in order to obtain the linear velocity and acceleration of the reference point, and ultimately decimated in order to obtain 100 samples per experiment. The total number or samples were thus $n_s' = 1200$.

The x and y coordinates of the marker were selected as *EE* free coordinates $\boldsymbol{\zeta}_f$. Accordingly, the experimental value of the controlled coordinates $\boldsymbol{\zeta}_c^\star$ were determined, for all t, as the numerical solution of the non-linear problem defined by:

$$\begin{cases} l_1(\boldsymbol{\zeta}_f^\star, \boldsymbol{\zeta}_c) - l_{1,0}^\star = 0 \\ \vdots \\ l_n(\boldsymbol{\zeta}_f^\star, \boldsymbol{\zeta}_c) - l_{n,0}^\star = 0 \end{cases} \tag{4.76}$$

where $l_i(\boldsymbol{\zeta}_f, \boldsymbol{\zeta}_c)$ is calculated as in Eq. (2.15), $\boldsymbol{\zeta}_f^\star$ contains the measured free coordinates, and $l_{i,0}^\star$ is the measured i-th cable length. Then, the *EE* twist and its time derivative were determined as (cf. Eqs. (2.52) and (2.59)):

$$\mathbf{v} = \boldsymbol{\Xi}_l^\perp \dot{\boldsymbol{\zeta}}_f, \qquad \dot{\mathbf{v}} = \dot{\boldsymbol{\Xi}}_l^\perp \dot{\boldsymbol{\zeta}}_f + \boldsymbol{\Xi}_l^\perp \ddot{\boldsymbol{\zeta}}_f \tag{4.77}$$

Finally, by collecting all experiment data, the *TLS* solution of the identification problem in Eq. (4.54) was obtained as:

$$\mathbf{X} = \begin{bmatrix} 1 \\ {}^{P}\mathbf{s}\prime \\ {}^{P}\check{\mathbf{I}}'_{P} \end{bmatrix} = \begin{bmatrix} 1 \\ 0.002 \\ -0.0025 \\ 0.1984 \\ 0.0569 \\ 0.0627 \\ 0.0314 \\ 2e-4 \\ -2e-4 \\ 1.6e-4 \end{bmatrix}, \qquad \boldsymbol{\sigma}_{\%X} = \begin{bmatrix} 0 \\ 4.32 \\ 3.09 \\ 0.15 \\ 0.64 \\ 0.31 \\ 1.76 \\ 80.5 \\ 99.4 \\ 136 \end{bmatrix}, \qquad \sigma_W = 0.536, \qquad C = 8.36 \tag{4.78}$$

Additional experiments were conducted by measuring the full *EE* pose during free motion, by means of additional markers mounted onto the platform. Thus, the use of Eqs. (4.76) and (4.77) was avoided, since pose data were complete, and the *EE* twist and its derivatives could be obtained by numerically differentiating the pose. The *TLS* solution of the identification problem in Eq. (4.54) corresponding to these data was obtained as:

$$\mathbf{X}_{full} = \begin{bmatrix} 1 \\ 0.0016 \\ -0.0025 \\ 0.2 \\ 0.0563 \\ 0.0627 \\ 0.0322 \\ 4.8e-5 \\ -2.5e-4 \\ 3.4e-4 \end{bmatrix}, \qquad \boldsymbol{\sigma}_{\%X,full} = \begin{bmatrix} 0 \\ 2.43 \\ 0.92 \\ 0.05 \\ 0.14 \\ 0.16 \\ 0.46 \\ 107 \\ -22 \\ 23 \end{bmatrix}, \qquad \sigma_W = 0.552, \qquad C = 10.26 \tag{4.79}$$

Both experiments show cost function values very close to the absolute minimum, thus both of them provide optimal results. The relative deviation of practically non-zero parameters are well below the 5% threshold for acceptability. The only parameters estimated with very high relative standard deviations are the off-diagonal elements of the inertia matrix, which have indeed very low magnitude. These parameters were then disregarded according to identification best practice, since they are *non-essential*. The identification problems were then modified by removing the column of $\mathbf{W}$ corresponding to these parameters, but the value of previously well-identified parameters, as well as their relative standard deviations, changed negligibly.

By comparing the identification results of experiments with partial and full pose measurements, the following can be noticed:

- first, the accuracy of the results did not practically change: the same parameters are well identified with relative standard deviations below 5%;
- second, the values of most parameters in $\mathbf{X}_{full}$ and $\mathbf{X}$ are comparable; if we compute the percentage relative error between $\mathbf{X}_{full}$ and $\mathbf{X}$ as:

$$\Delta X_{i,\%} = 100 \left\| \frac{X_{i,full} - X_i}{X_i} \right\| \% \tag{4.80}$$

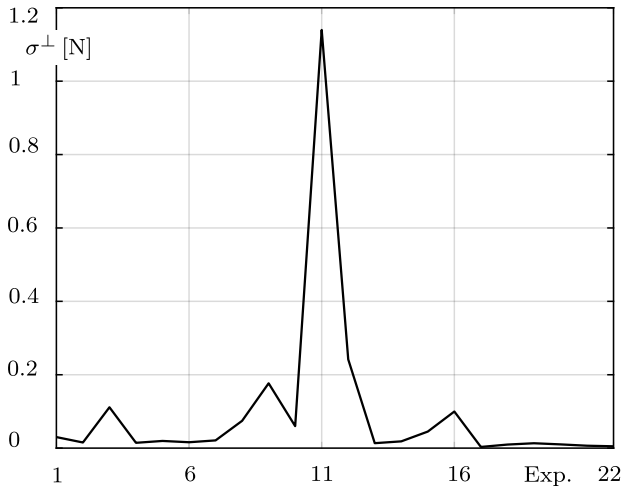


Figure 4.4: Internal dynamics standard deviation in cross-validating experiments

and we compare them with $\boldsymbol{\sigma}_{\%X}$:

$$\boldsymbol{\sigma}_{\%X} = \begin{bmatrix} 0 \\ 4.32 \\ 3.09 \\ 0.15 \\ 0.64 \\ 0.31 \\ 1.76 \\ 80.5 \\ 99.4 \\ 136 \end{bmatrix}, \qquad \Delta\mathbf{X}_{\%} = \begin{bmatrix} 0 \\ 21.14 \\ 0.01 \\ 0.76 \\ 1.11 \\ 0.04 \\ 2.77 \\ 76.53 \\ 22.18 \\ 109.66 \end{bmatrix} \tag{4.81}$$

we can conclude that the two results are equivalent in practice, since the relative errors between $\mathbf{X}_{full}$ and $\mathbf{X}$ are of the same order of magnitude of $\mathbf{X}$ standard deviations. The only parameter which is relatively different by comparison is the x coordinate of the center of mass. On the other hand, the absolute difference of these components is 0.4mm, which is negligible in practice of an *EE* whose dimension is of the decimeter order.

In order to assess whether identification results are generally valid, 22 cross-validation free-motion experiments were conducted in random configurations (different from the ones used for identification), and the *EE* full pose was measured. Model error $\boldsymbol{\varepsilon} = \mathbf{WX}$ was calculated for each experiment, with matrix $\mathbf{W}$ computed according to the measured data and $\mathbf{X}$ as in Eq. (4.78), and internal dynamics standard deviation $\sigma^{\perp}$ was calculated as:

$$\sigma^{\perp} = \sqrt{\frac{\boldsymbol{\varepsilon}^T \boldsymbol{\varepsilon}}{n_{cv}}} \tag{4.82}$$

with n_{cv} being the number of samples in the cross-validation experiments, $n_{cv} = 1001$. The values of $\sigma^{\perp}$ for the cross-validation experiments are reported in Fig. 4.4. The order of magnitude $\sigma^{\perp}$ for each experiment is near the one of σ_W in Eq. (4.78) (only one experiment has $\sigma^{\perp} > \sigma_W$), thus identification results are validated.

We can conclude that the identification method proposed is both theoretically and practically effective in determining *EE* dynamic parameters with low errors. In the future, we will consider different optimal excitation strategies that can be fully automatic

and do not require manual intervention. Additionally, the external measuring device will be substituted with a pose estimation strategy which employs low-cost proprioceptive sensors.

4.3 Initial Cable Length Estimation

A major issue in the practical use of underactuated *CDPRs* is the estimation of the *EE* initial-pose. When the machine is switched on in a generic start-up condition, the *EE* pose is generally unknown, but its knowledge is fundamental for any subsequent operation. In order to directly measure the *EE* pose, external measurement devices such as laser trackers [91] or high-resolution cameras [92] can be employed. On the other hand, an indirect estimation of the pose can be performed by measuring some of the robot's internal joint variables, followed by the solution of the direct kinematic problem [93, 94]. When this approach is used for pose estimation in start-up conditions, the solution to this problem is sometimes referred to as *self-calibration* [95] or *internal-calibration* [96] of the *EE* initial-pose.

In [95], a self-calibration procedure for initial-pose estimation of a 2-*DoF* 4-cable over-constrained robot is proposed, which is based on cable tension and length increment measurements. The proposed method relies on the over-constrained nature of the robot. In [97] a two-stage calibration procedure for generic over-constrained *CDPRs* is introduced, aiming at both optimizing robot static parameters and determining the initial-pose of the *EE*. Ref. [96] shows how to perform initial-pose estimation by means of a manual self-calibration procedure for over-constrained robots, only relying on cable length increment measurements.

This Section extends the method introduced in [96] by proposing an automatic procedure to estimate the initial-pose of a generic suspended underactuated *CDPR*, e.g. a 6-*DoF CDPR* actuated by $n < 6$ cables, that only relies on incremental measurements of cable lengths and orientations. An automatic data acquisition procedure is exploited in order to reliably and autonomously collect the information required for estimation purposes.

The initial-pose estimation is formulated as a non-linear least square optimization problem (*NLLS*), which aims at minimizing the error of an extended forward geometrico-static problem including swivel angles constraints. The problem tentative solution is generated automatically according to a data acquisition algorithm. In Section 4.3.1, the extended forward geometrico-static model of an underactuated *CDPR* which includes swivel pulley angles measurement is developed. In Sections 4.3.2 and 4.3.3, the *NLLS* optimization problem is formulated and the data acquisition algorithm employed for its solution is discussed. Finally, simulation and experiments are presented.[5]

4.3.1 Extended forward geometrico-static problem

The aim of the forward geometrico-static problem, as analyzed in Sec. 2.4.1, is to determine the *EE* pose, once the value of a suitable set of the robot's internal joint variables is known. In the case that only cable length measurements are available, the problem can be formulated as in Eq. (2.86), resulting in a completely determined system of nonlinear equations, which thus can be numerically solved by using nonlinear

[5] The content of this Section was published in [44].

solvers. Alternatively, if redundant measurements are available, one can (possibly) neglect the static constraint equations and replace them with additional kinematic constraint equations that are explicitly dependent on the measured variables [93]. This approach leads to a system of equations that, depending on the number and the type of redundant measurements, can be completely determined or overdetermined (the system is never underdetermined, because static equilibrium constraints can always be accounted for).

In this Section, we assume that both cable lengths and swivel pulley angles can be measured. We thus formulate an extended forward geometrico-static problem by considering both $2n$ kinematic constraints, derived from Eqs. (2.6) and (2.15), and the $6-n$ equations deriving from the static constraint of Eq. (2.83), thus leading to an overdetermined system of $6+n$ equations in 6 unknowns $\boldsymbol{\zeta}$. By letting:

$$\mathbf{F}_1(\boldsymbol{\zeta}) = \begin{bmatrix} \sigma_1(\boldsymbol{\zeta}) - \sigma_1{}^\star \\ \vdots \\ \sigma_n(\boldsymbol{\zeta}) - \sigma_n{}^\star \end{bmatrix}, \qquad \mathbf{F}_2(\boldsymbol{\zeta}) = \begin{bmatrix} l_1(\boldsymbol{\zeta}) - l_1{}^\star \\ \vdots \\ l_n(\boldsymbol{\zeta}) - l_n{}^\star \end{bmatrix}, \qquad \mathbf{F}_3(\boldsymbol{\zeta}) = \mathbf{f}^\perp(\boldsymbol{\zeta}) \tag{4.83}$$

where $l_i(\boldsymbol{\zeta})$ and $\sigma_i(\boldsymbol{\zeta})$ are calculated as in Eq. (2.15) and (2.6), respectively, and $l_i{}^\star$ and $\sigma_i{}^\star$ are the measured length of the i-th cable and swivel angle of the i-th pulley. The extended forward geometrico-static problem in then formulated as:

$$\mathbf{F}(\boldsymbol{\zeta}) = \begin{bmatrix} \mathbf{F}_1(\boldsymbol{\zeta}) \\ \mathbf{F}_2(\boldsymbol{\zeta}) \\ \mathbf{F}_3(\boldsymbol{\zeta}) \end{bmatrix} = \mathbf{0}_{6+n} \tag{4.84}$$

Equation (4.84) admits a fairly simple analytical formulation of its first-order differentiation,, as described in Section 4.3.2, which is needed for the solution rapid computation.

4.3.2 Initial-pose estimation problem

In this section, the initial-pose estimation problem will be formulated by extending the work presented in [96]. If the underactuated *CDPR* is equipped with incremental measurement devices on motors and swivel axes, i.e. incremental encoders, cable lengths and swivel angles at a generic pose $\boldsymbol{\zeta}_i$ can be measured relatively to their initial values σ_i^0 and l_i^0 at pose $\boldsymbol{\zeta}_0$, namely:

$$\sigma_i^\star = \sigma_i^0 + \Delta\sigma_i^\star \tag{4.85}$$

$$l_i^\star = l_i^0 + \Delta l_i^\star \tag{4.86}$$

While $\Delta\sigma_i^\star$ and $\Delta l_i^\star$ are measures provided by the encoders, σ_i^0 and l_i^0 are generally unknown and are the objective of the self-calibration procedure. The extended forward geometrico-static problem in Eq. (4.84) can thus be expressed as:

$$\mathbf{F}(\boldsymbol{\sigma}^0, \mathbf{l}^0, \boldsymbol{\zeta}) = \mathbf{0}_{6+n} \tag{4.87}$$

where $\boldsymbol{\sigma}^0 = [\sigma_1^0, \cdots, \sigma_n^0]^T$ and $\mathbf{l}^0 = [l_1^0, \cdots, l_n^0]^T$. This problem has $6+n$ equations and $6+2n$ unknowns ($\boldsymbol{\sigma}^0$, $\mathbf{l}^0$, $\boldsymbol{\zeta}$), and it has generally an infinite number of solutions.

However, by assuming that n_m different measurement sets are available, that is:

$$\begin{aligned} \sigma_{i,k}^\star &= \sigma_i^0 + \Delta\sigma_{i,k}^\star \\ l_{i,k}^\star &= l_i^0 + \Delta l_{i,k}^\star \end{aligned} \qquad k = 1, \cdots, n_m \tag{4.88}$$

the following system of equations is obtained:

$$\mathbf{G}(\mathbf{X}) = \mathbf{G}(\boldsymbol{\sigma}^0, \mathbf{l}^0, \boldsymbol{\zeta}_1, \cdots, \boldsymbol{\zeta}_{n_m}) = [\mathbf{F}(\boldsymbol{\sigma}^0, \mathbf{l}^0, \boldsymbol{\zeta}_1); \cdots; \mathbf{F}(\boldsymbol{\sigma}^0, \mathbf{l}^0, \boldsymbol{\zeta}_{n_m})] = \mathbf{0}_{(6+n)n_m} \tag{4.89}$$

where $\mathbf{X} = [\boldsymbol{\sigma}^{0T}, \mathbf{l}^{0T}, \boldsymbol{\zeta}_1^T, \cdots, \boldsymbol{\zeta}_{n_m}^T]^T \in \mathbb{R}^{6n_m+2n}$.

The system (4.89) has a total of $(6+n)n_m$ equations and $6n_m+2n$ unknowns. Thus, if $n_m > 2$, the initial-pose estimation problem is overdetermined and can be formulated as a non-linear least-square optimization:

$$\mathbf{X}_{\text{opt}} = \arg\min_{\mathbf{X}} \|\mathbf{G}(\mathbf{X})\|^2 \tag{4.90}$$

This problem can be solved by employing numerical techniques, such as the Levenberg-Marquardt algorithm. The efficient solution of Eq. (4.90) relies on a reasonable tentative solution $\mathbf{X}_{\text{guess}}$ (see Section 4.3.3), and an analytical formulation of the Jacobian matrix of Eq.(4.89). While the former is fundamental for both the solution accuracy and the algorithm rapidity, the latter is critical only in terms of computational time.

The Jacobian $\partial\mathbf{G}/\partial\mathbf{X}$ can be expressed analitically as:

$$\frac{\partial \mathbf{G}}{\partial \mathbf{X}} = \begin{bmatrix} -\mathbf{I}_{n\times n} & \mathbf{0}_{n\times n} & \partial\boldsymbol{\sigma}(\boldsymbol{\zeta}_1)/\partial\boldsymbol{\zeta} & \mathbf{0}_{n\times 6} & \cdots & \mathbf{0}_{n\times 6} \\ \mathbf{0}_{n\times n} & -\mathbf{I}_{n\times n} & \partial\mathbf{l}(\boldsymbol{\zeta}_1)/\partial\boldsymbol{\zeta} & \mathbf{0}_{n\times 6} & \cdots & \mathbf{0}_{n\times 6} \\ \mathbf{0}_{(6-n)\times n} & \mathbf{0}_{(6-n)\times n} & \partial\mathbf{F}_3(\boldsymbol{\zeta}_1)/\partial\boldsymbol{\zeta} & \mathbf{0}_{(6-n)\times 6} & \cdots & \mathbf{0}_{(6-n)\times 6} \\ -\mathbf{I}_{n\times n} & \mathbf{0}_{n\times n} & \mathbf{0}_{n\times 6} & \partial\boldsymbol{\sigma}(\boldsymbol{\zeta}_2)/\partial\boldsymbol{\zeta} & \cdots & \mathbf{0}_{n\times 6} \\ \mathbf{0}_{n\times n} & -\mathbf{I}_{n\times n} & \mathbf{0}_{n\times 6} & \partial\mathbf{l}(\boldsymbol{\zeta}_2)/\partial\boldsymbol{\zeta} & \cdots & \mathbf{0}_{n\times 6} \\ \mathbf{0}_{(6-n)\times n} & \mathbf{0}_{(6-n)\times n} & \mathbf{0}_{(6-n)\times 6} & \partial\mathbf{F}_3(\boldsymbol{\zeta}_2)/\partial\boldsymbol{\zeta} & \cdots & \mathbf{0}_{(6-n)\times 6} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ -\mathbf{I}_{n\times n} & \mathbf{0}_{n\times n} & \mathbf{0}_{n\times 6} & \mathbf{0}_{n\times 6} & \cdots & \partial\boldsymbol{\sigma}(\boldsymbol{\zeta}_{n_s})/\partial\boldsymbol{\zeta} \\ \mathbf{0}_{n\times n} & -\mathbf{I}_{n\times n} & \mathbf{0}_{n\times 6} & \mathbf{0}_{n\times 6} & \cdots & \partial\mathbf{l}(\boldsymbol{\zeta}_{n_s})/\partial\boldsymbol{\zeta} \\ \mathbf{0}_{(6-n)\times n} & \mathbf{0}_{(6-n)\times n} & \mathbf{0}_{(6-n)\times 6} & \mathbf{0}_{(6-n)\times 6} & \cdots & \partial\mathbf{F}_3(\boldsymbol{\zeta}_{n_s})/\partial\boldsymbol{\zeta} \end{bmatrix} \tag{4.91}$$

where $\boldsymbol{\sigma}(\boldsymbol{\zeta}_k) = [\sigma_1(\boldsymbol{\zeta}_k); \cdots; \sigma_n(\boldsymbol{\zeta}_k)]$ and $\mathbf{l}(\boldsymbol{\zeta}_k) = [l_1(\boldsymbol{\zeta}_k); \cdots; l_n(\boldsymbol{\zeta}_k)]$ and, accounting for Eqs. (2.25), (2.27), and (2.101) :

$$\frac{\partial \boldsymbol{\sigma}}{\partial \boldsymbol{\zeta}} = \mathbf{J}_\sigma^T, \qquad \frac{\partial \mathbf{l}}{\partial \boldsymbol{\zeta}} = \mathbf{J}_l^T, \qquad \frac{\partial \mathbf{F}_3}{\partial \boldsymbol{\zeta}} = \frac{\partial \mathbf{f}^\perp}{\partial \boldsymbol{\zeta}} = \boldsymbol{\Xi}_l^{\perp T} \left[(\mathbf{K}+\mathbf{E})\,\mathbf{D} - \mathbf{F}\right] \tag{4.92}$$

4.3.3 Data acquisition algorithm

It is beyond the scope of this work to determine an optimal data acquisition algorithm. However, a practical one, which enables autonomous and safe operation of the *CDPR* during calibration, is provided hereafter. For this aim, cable tensions or alternatively motor torques are assumed to be measurable or at least estimated, and actively controlled by a suitable feedback system.

In the instant the robot is switched on, its pose is generally unknown. It is possibly unsafe to start the self-calibration process in this configuration. Then, it is useful to pre-determine a safe start configuration, in which every cable is taut and sufficiently long, so that it can be coiled and uncoiled, and the *EE* may attain different poses. By assigning a start cable tension vector $\boldsymbol{\tau}_0$, the static problem (2.82) can be solved for $\boldsymbol{\zeta}$ as a non-linear system of six equations in six unknowns. Because of the non-linearity of the problem, a finite set of real solutions can be determined: this calculation may

be done off-line and just once, i.e. during robot parameter calibration. Only stable solutions [15] among the possibly many available should be considered. Additionally, $\boldsymbol{\tau}_0$ may be selected so that only one stable solution exist.

In the following, we will consider a start cable tension vector leading to a unique stable solution of problem (2.82). Accordingly, a (computed) start pose $\boldsymbol{\zeta}_{0\text{comp}}$ is unambiguously determined, as well as start cable lengths $\mathbf{l}^0_{\text{comp}}$ and swivel angles $\boldsymbol{\sigma}^0_{\text{comp}}$. The real start pose $\boldsymbol{\zeta}_0$ attained by the *CDPR* can be fairly different from the ideal one $\boldsymbol{\zeta}_{0\text{comp}}$, and its determination is the aim of the self-calibration procedure. A maximum cable tension vector $\boldsymbol{\tau}_m$ should be set as well. The data acquisition algorithm objective is to ensure that every *DoF* of the *EE* is varied during measurements, so that problem (4.90) is always well conditioned. The procedure workflow can be summarized as follows.

1. *Start phase*: command the *CDPR* so that cable tensions (or motor torques) quasi-statically reach the assigned value $\boldsymbol{\tau}_0$. When $\boldsymbol{\tau}_0$ is reached and static conditions are attained[6], the j-th actuator is assigned an incremental cable tension (or motor torque) set-point, starting from $j = 1$. The change in a single actuator set-point ensures that the pose of the end-effector is different at any iteration, thus being effective, as well as practical and easy to implement;

2. *Tensioning phase*: quasi-statically move the *CDPR* by assigning $n_m/(2n)$ positive increments of magnitude $\Delta\tau = 2n(\tau_{j,m} - \tau_{j,0})/n_m$ to the tension set-point of the j-th actuator, namely $\tau_{j,k} = \tau_{j,k-1} + \Delta\tau$, where $\tau_{j,0}$ is the j-th component of $\boldsymbol{\tau}_0$. After each assignment k, the *CDPR EE* could possibly oscillate during the transition. When static conditions are attained, record measurements $\Delta\sigma^\star_{i,k}$ and $\Delta l^\star_{i,k}$, for $i = 1, \cdots, n$;

3. *Detensioning phase*: assign $n_m/(2n)$ negative increments of magnitude $\Delta\tau$ to the tension set-point of the j-th actuator, namely $\tau_{j,k} = \tau_{j,k-1} - \Delta\tau$. When static conditions are attained, record measurements $\Delta\sigma^*_{i,k}$ and $\Delta l^*_{i,k}$, for $i = 1, \cdots, n$. During the detensioning phase, the robot follows exactly the same cable tension (motor torques) set-points as in the tensioning phase: on a real machine, due to repeatability errors, this could lead to different cable lengths and swivel angles, which are possibly useful in the calibration procedure in order to minimize the repeatability error of the robot. n_m/n measurement sets are thus obtained by varying a single actuator set-point;

4. *Increment phase*: $j = j + 1$; if $j \leq n$ then go to point 2, otherwise the algorithm is finished because λ measurement sets have been recorded.

The initial guess for the solution of problem (4.90) is computed as:

$$\mathbf{X}_{\text{guess}} = \left[\boldsymbol{\sigma}^0_{\text{comp}};\ \ \mathbf{l}^0_{\text{comp}};\ \ \boldsymbol{\zeta}_{1,\text{comp}};\ \ \cdots;\ \ \boldsymbol{\zeta}_{n_m,\text{comp}}\right] \tag{4.93}$$

where $\boldsymbol{\zeta}_{k,\text{comp}}$, $k = 1, \ldots, n_m$, can be evaluated by solving the static problem (2.82) with assigned tension $\boldsymbol{\tau}_k$. Finally, by employing $\mathbf{X}_{\text{guess}}$, $\Delta\boldsymbol{\sigma}^\star_k = [\Delta\sigma^\star_{1,k} \cdots \Delta\sigma^\star_{n,k}]^T$ and

[6] Notice that cable tensions are only used to lead the platform to poses where the robot is stable and kinematic measures can be accurately performed. They have no role in the optimization problem solution, since they do not appear as variables in Eq. (4.90). They may be affected by appreciable errors without compromising the procedure, whereas platform stability plays a key role in the data acquisition process.

Table 4.3: Experiments results

i	1	2	3	4	5
$\boldsymbol{\zeta}_0$[mm, °]	$\begin{bmatrix} -1314 \\ 900 \\ -355 \\ -5.0 \\ -4.8 \\ -32.0 \end{bmatrix}$	$\begin{bmatrix} -1345 \\ 922 \\ -322 \\ -4.0 \\ -3.6 \\ -31.9 \end{bmatrix}$	$\begin{bmatrix} -1348 \\ 917 \\ 310 \\ -4.1 \\ -3.4 \\ -32.1 \end{bmatrix}$	$\begin{bmatrix} -1349 \\ 917 \\ -310 \\ -4.1 \\ -3.4 \\ -32.0 \end{bmatrix}$	$\begin{bmatrix} -1344 \\ 913 \\ -314 \\ -4.2 \\ -3.6 \\ -32.0 \end{bmatrix}$
$\boldsymbol{\zeta}_{0,\text{opt}}$[mm, °]	$\begin{bmatrix} -1324 \\ 906 \\ -361 \\ -4.5 \\ -4.9 \\ -32.1 \end{bmatrix}$	$\begin{bmatrix} -1347 \\ 913 \\ -325 \\ -4.2 \\ -4.0 \\ -32.0 \end{bmatrix}$	$\begin{bmatrix} -1357 \\ 895 \\ -310 \\ -5.0 \\ -3.6 \\ -32.0 \end{bmatrix}$	$\begin{bmatrix} -1359 \\ 914 \\ -307 \\ -4.2 \\ -3.5 \\ -31.9 \end{bmatrix}$	$\begin{bmatrix} -1348 \\ 918 \\ -315 \\ -4.0 \\ -3.9 \\ -31.9 \end{bmatrix}$
$\|\mathbf{e}_p\|$ [mm]	13.4	9.9	23.4	11.5	5.7
$\|\mathbf{e}_e\|$ [°]	0.5	0.4	0.9	0.2	0.4

$\Delta\mathbf{l}_k^\star = [\Delta l_{1,k}^\star \cdots \Delta l_{n,k}^\star]^T$ for $k = 1, \cdots, n_m$, it is possible to determine $\mathbf{X}_{\text{opt}}$ as a solution of (4.90). Ideally, $\boldsymbol{\sigma}_{\text{opt}}^0$ and $\mathbf{l}_{\text{opt}}^0$ should converge to $\boldsymbol{\sigma}^0$ and $\mathbf{l}^0$, respectively.

4.3.4 Experimental Results

The proposed data acquisition strategy and calibration method was tested on a prototype. Swivel angles were measured by 16-bit incremental encoders, mounted directly on the swivel axes of pulleys, whereas cable lengths were estimated by using 20-bit incremental encoders on each motor axis and a kinematic model of the winch. Swivel pulleys were manufactured by *FDM* technology, thus limited, but not negligible, errors in their geometry and elasticity exist. Cables were coiled on IPAnema winches [98]. Clearance and elasticity in the winch components, as well as cable elasticity itself, are possible sources of error in the estimation of cable lengths. Robot geometric and inertial parameter for this experiments can be found in App. B.2.

In order to measure the real pose of the platform during experiments, 8 VICON Motion Capture Systems cameras were employed to track the position of 4 markers mounted on the robot platform.The accuracy of the measure is ±0.2mm for each dimension of the marker (x, y, z), according to manufacturer specifications. In the end, the position of the reference point and the platform orientation were reconstructed from the recorded position of each marker.

Because of the lack of force sensors in the robot set-up, motor torques were employed instead of cable tensions for the implementation of the algorithm presented in Section 4.3.3. The start tension vector and maximum cable tensions were set to $\boldsymbol{\tau}_0 = \begin{bmatrix} 40.0 & 40.0 & 40.0 \end{bmatrix}^T$ N and $\boldsymbol{\tau}_m = \begin{bmatrix} 80.0 & 80.0 & 80.0 \end{bmatrix}^T$ N, respectively, and converted in motor torques according to static equilibrium of the cable transmissions. In the end,

$n_s = 60$ was chosen as a trade-off between accuracy of the initial-pose estimation and data-acquisition speed.

The results of five experiments are reported in Table 4.3, where $\boldsymbol{\zeta}_0$ is the real starting pose, as measured by the motion tracking system, $\boldsymbol{\zeta}_{0,\text{opt}}$ is the estimated starting pose resulting from the solution of the problem (4.90), $\|\mathbf{e}_p\| = \|\boldsymbol{\zeta}_0 - \boldsymbol{\zeta}_{0,\text{opt}}\|$ is the reference position error norm and $\|\mathbf{e}_\epsilon\| = \|\boldsymbol{\epsilon}_0 - \boldsymbol{\epsilon}_{0,\text{opt}}\|$ is the norm of the error of the orientation parameters. Positions are expressed in millimeters and angles in degrees. The execution of the calibration procedure required, on average, 4 min for the data acquisition procedure, and 2.5 s for the initial-pose estimation, which was implemented in Matlab.

During experiments, it was observed that the orientation of the swivel pulley axes plays a crucial role for the conditioning of problem (4.90). Pulley orientations were set in order to achieve the best possible results with the robot architecture at hand, but are not optimal. A maximum reference position error of 23.4mm and a maximum orientation error of 0.9° are satisfactory, considering the modelling simplifications and the hardware used for experimentation. The *UACDPR* prototype had a workspace of roughly 6m^3 and most of the components were made of 3D-printed plastic. In addition, swivel pulleys were not statically balanced: accordingly, when the swivel pulley center of mass does not belong to the swivel axis, the cable direction cannot belong to the pulley plane because it has to counteract the pulley weight, which is unmodelled in this thesis. This effect was observed in our lab to be non negligible, especially when low cable tensions are attained.

Finally, a good knowledge of the geometric parameters of the pulley transmission, which correlates the measured swivel-pulley rotations to *EE* displacements, may have a non-negligile impact in the statics of the *UACDPR* [48], but geometric-parameter accurate determination is hard if swivel pulleys are mounted on a large structure. A better swivel pulley design and a proper calibration of the system parameters which influence the extended forward geometrico-static problem are expected to provide better results than those reported in this Section, and they will be investigated in the future. Preliminary experiments in our laboratory showed that the application of standard kinematic calibration models [99] are not sufficient for the determination of pulley geometric parameters, which are reasonably important if rotary encoders on swivel pulley axes are employed for estimating the *EE* pose. Thus, a novel technique dedicated to their determination is in order.

Chapter 5

Trajectory Planning

When planning the trajectory of a completely-actuated manipulator, we may define both the geometric path of a reference point on the end-effector and its orientation [100]. If the end-effector is underactuated, however, only a subset $\boldsymbol{\zeta}_c$ of its generalized coordinates, namely the controlled coordinates, may be assigned [101]. It is convenient to consider a parametric representation of the path to track, such as $\boldsymbol{\zeta}_c = \boldsymbol{\zeta}_c(u(t))$. We refer to the parameter $u(t)$ as the *motion law*, which is a function of time, with initial and final conditions $u(0) = 0$ and $u(T) = 1$, where T is the trajectory execution time. The composition $(\boldsymbol{\zeta}_c \circ u) = \boldsymbol{\zeta}_c(u(t))$ is what is usually referred to as the *trajectory*. However, if an arbitrary motion is prescribed for $\boldsymbol{\zeta}_c$, the constraint deficiency on the end-effector motions leads to the impossibility of bringing the system at rest in a prescribed time, since the free coordinates $\boldsymbol{\zeta}_f$ evolve according to the system internal dynamics. In addition, the behavior of the system may not be stable, that is, unbounded oscillatory motion of the end-effector may arise.

In this Chapter, the problem of trajectory planning of *UACDPR* is analyzed, and different planning techniques proposed. First, trajectory planning techniques aiming at reducing *EE* oscillation are discussed, and two solutions which considers the use of *Multi-Mode Input Shaping* and additionally *Dynamic Scaling* are laid out. Then, the problem of rest-to-rest trajectory planning, which system theory defines as a transition problem between stationary setpoints [102], is considered. A solution procedure, which accounts for constraints on motion time and path geometry, is put forward.

5.1 Oscillation-Limiting Trajectory Planners

When planning a trajectory of duration T of the actuated coordinates $\boldsymbol{\zeta}_c$, and the geometric path to follow $\boldsymbol{\zeta}_c(u(t))$ is assigned, only the motion law needs to be designed. In the case of line-segment paths, which are the ones considered in this Section for the sake of simplicity, the trajectory connecting two set-points $\boldsymbol{\zeta}_{c,s}$ and $\boldsymbol{\zeta}_{c,f}$ is:

$$\boldsymbol{\zeta}_c(t) = \boldsymbol{\zeta}_{c,s} + (\boldsymbol{\zeta}_{c,f} - \boldsymbol{\zeta}_{c,s})\, u(t) \tag{5.1}$$

In order to reduce *EE* oscillations, the authors of [28–30] proposed to apply Input Shaping (*IS* in short) to 2- and 3-cable *UACDPRs*, as well as to a simplified model of 4-cable *UACDPR*. *IS* is a well-known and simple approach dedicated to the reduction of oscillations of second-order linear dynamic systems [103]. In order to compute *UACDPRs* natural oscillation frequencies, the *EE internal dynamics* [45] needs to be derived first,

and expressed in terms of a minimal set of *EE* residual *DoFs* next. Finally, natural frequencies are determined by linearizing, about an equilibrium configuration, the *EE* internal dynamics with respect to the *EE* residual *DoFs*, and by solving the resulting eigenproblem. The authors of [28] derived the single configuration-dependent natural oscillation frequency of a planar 3-DoF 2-cable robot, by intuitively selecting the platform orientation as the *EE* residual *DoF*, whereas in [29] the same technique was employed for a spatial 6-*DoF* 3-cable system, where *ZYX* Tait-Bryan angles where chosen as residual *DoFs*. In both cases, the translational and rotational mechanical equilibria of the *EE* could be decoupled, which resulted in a mathematically simpler internal dynamics formulation and linearization. This was not the case, instead, for the 6-*DoF* 4-cable manipulator considered in [30], where the authors determined the system natural frequencies with a method similar to the one developed in [28], by approximating the 6-*DoF* robot with two 3-*DoF* planar sub-systems and selecting the orientation of these sub-systems platforms as residual *DoFs*. In fact, because of the intrinsic coupling of rotational and translational equilibria of 6-*DoF UACDPRs* with more than 3 cables, it is not straightforward to derive and linearize manipulator internal dynamic equations and select residual *EE DoFs* (1-*DoF* for 5-cable robots and 2-*DoFs* for 4-cable robots): representation singularities may arise during calculations.

In this Section, Multi-Mode Zero-Vibration Input Shaping (*MMZVIS*) is applied to a generic 4-cable *UACDPR* thanks to the general model outlined in Section 4.1: our technique provides a solution for the selection of *EE* residual *DoFs* so that representation singularities are always avoided in the formulation and linearization of *EE* internal dynamics and the subsequent natural frequency computation. The approximated model employed in [30] demonstrated its effectiveness only for slowly varying trajectories (the *EE* mean linear velocity module was 0.4 m/s), whereas the more accurate model proposed in this thesis is able to effectively reduce oscillations even if high speed trajectories are considered (the *EE* mean linear velocity module is larger than 1.2 m/s). Furthermore, the combination of *MMZVIS* and Dynamic Scaling (*DS*) of the reference trajectory is considered in order to robustify oscillation-limiting trajectories against the variation of robot natural oscillation frequencies, which are configuration dependent and thus variable during a trajectory execution.[1]

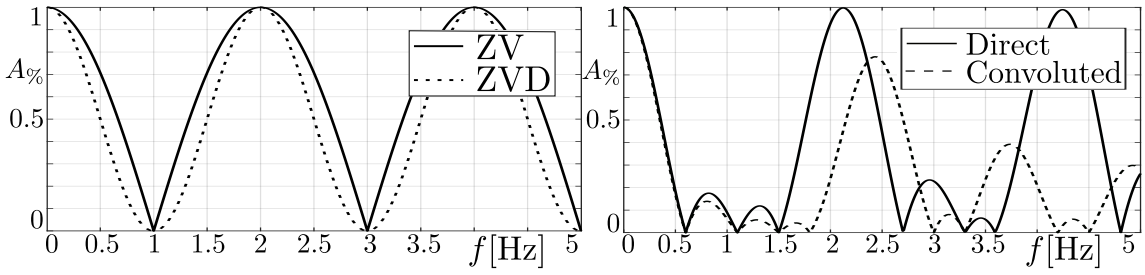
IS practical implementation requires the convolution of a series $S(t)$ of impulses, called the *input shaper*, with a reference signal. By denoting the convolution operator with $*$, the input shaping of the trajectory in Eq. (5.1) is given by:

$$\boldsymbol{\zeta}_c(t) * S(t) = \boldsymbol{\zeta}_{c,s} + (\boldsymbol{\zeta}_{c,f} - \boldsymbol{\zeta}_{c,s})\,(u(t) * S(t)), \quad S(t) = \sum_{i=1}^{k} A_i \delta_i (t - t_i) \tag{5.2}$$

where $\delta_i(t = t_i) = 1$, $\delta_i(t \neq t_i) = 0$, A_i is the impulse amplitude, t_i is the time at which it occurs, and k is the number of impulses. The pairs (A_i, t_i) can be determined by setting to zero the amplitude $A_{\%}(f)$ of the Fourier trasform of $S(t)$, for an assigned frequency f and k impulses (*Zero-Vibration* or *ZV IS* [103], Fig. 5.1a):

$$A_{\%}(f) = \sqrt{\left(\sum_{i=1}^{k} A_i \cos(2\pi f t_i)\right)^2 + \left(\sum_{i=1}^{k} A_i \sin(2\pi f t_i)\right)^2} = 0 \implies (A_i, t_i) \tag{5.3}$$

[1] The content of this Section was partly submitted to *IEEE Transaction on Robotics* and partly published in [46].



(a) ZV and ZVD IS for $f = 1$Hz

(b) Direct and Convoluted IS for $f_j = 0.6, 1.1, 1.5$Hz

Figure 5.1: Examples of Input-Shapers

By definition, an input shaper is able to eliminate oscillations at the given frequency f (and some multiples of it), and to reduce the amplitude of oscillation associated with every other frequency (see the graph ZV in Fig. 5.1a), but always introduces a delay in trajectory execution, which is quantifiable as $t_d = t_k$, the time location of the k-th impulse. If the shaper is supposed to eliminate the oscillations at μ frequencies $f_j\,(j = 1,\ldots,\mu)$, two techniques can be employed, which result in a different number of impulses and time delays [104]. The so-called *direct method* always uses the minimum number of impulses $k_{dir} = 1+\mu$, whereas the so-called *convolved method* leads to $k_{conv} = 2^{\mu}$. Usually, $t_{d,dir} < t_{d,conv} = \sum_{j=1}^{\mu} 1/(2f_j)$, but $A_{\%}$ is slightly higher when it is not zero [104] (see Fig. 5.1b for $f_j = 0.6, 1.1, 1.5$Hz, $\mu = 3$). On the practical side, a convolved *IS* is easy to determine, because it results from the convolution of μ *ZV* shapers (each one with 2 impulses), which can be computed analytically, whereas a direct *IS* has to be numerically computed by imposing Eq. (5.3) to be satisfied simultaneously at $f_1,\ldots,f_{\mu}$, which requires $1+\mu$ impulses (one impulse more for each additional frequency in the shaper). It should be noted that the minimum number of impulses of a direct shaper is always preferable for real-time implementation, since the amount of time required for the calculation of $u(t) * S(t)$, for each t, and thus trajectory computational complexity, increases with the number of impulses in the shaper.

5.1.1 Input-Shaping of a nominal trajectory

On a 4-cable *UACDPR*, whose model parameters are defined in App. C.1, three trajectories with equal start and final configurations, only differing in the choice of the motion law $u(t)$, are experimentally compared:

- the first motion law, which we call STD_T, is a standard trapezoidal velocity profile, with total transition time T, and αT acceleration and deceleration duration ($0 \le \alpha \le 0.5$):

$$u_{\mathrm{STD}_T}(t) = \begin{cases} \frac{(t/T)^2}{2\alpha(1-\alpha)}, & t < \alpha T \\ \frac{(-\alpha+2t/T)}{2(1-\alpha)}, & \alpha T \le t \le (1-\alpha)T \\ \frac{-2\alpha^2+2\alpha-1+t/T-(t/T)^2}{2\alpha(1-\alpha)}, & t > (1-\alpha)T \end{cases} \tag{5.4}$$

- the second motion law, which we call $\mathrm{STD}_T - \mathrm{IS}$, is the convolution of STD_T with a multi-mode zero-vibration input shaper:

$$u_{\mathrm{STD}_T-\mathrm{IS}}(t) = u_{\mathrm{STD}_T}(t) * S(t) \tag{5.5}$$

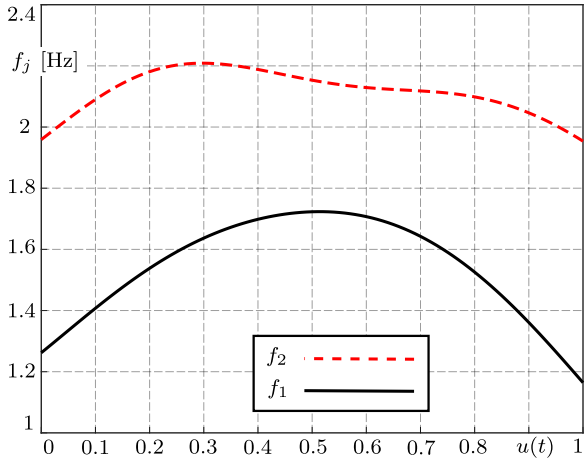


Figure 5.2: Modelled oscillation frequencies along $u(t)$.

the convolution with the input-shaper thus delays the total duration of the trajectory by ΔT;

- the last motion law, which we call $\text{STD}_{T+\Delta T}$, is a standard trapezoidal velocity profile, with total transition time $T+\Delta T$, and $\alpha(T+\Delta T)$ acceleration and deceleration duration.

Once a trajectory for the controlled coordinates is assigned, the evolution of the free coordinates when the system is following a prescribed trajectory must be evaluated. Free coordinates are computed by numerically integrating the system internal dynamics (2.68):

$$\mathbf{M}^{\perp}_{cP}\ddot{\boldsymbol{\zeta}}_c + \mathbf{M}^{\perp}_{fP}\ddot{\boldsymbol{\zeta}}_f + \mathbf{C}^{\perp}_{cP}\dot{\boldsymbol{\zeta}}_c + \mathbf{C}^{\perp}_{fP}\dot{\boldsymbol{\zeta}}_f + \mathbf{f}^{\perp} = \mathbf{0}_{\lambda\times 1} \tag{5.6}$$

after its state-form re-definition:

$$\mathbf{x} = \begin{bmatrix} \boldsymbol{\zeta}_f \\ \dot{\boldsymbol{\zeta}}_f \end{bmatrix} \tag{5.7}$$

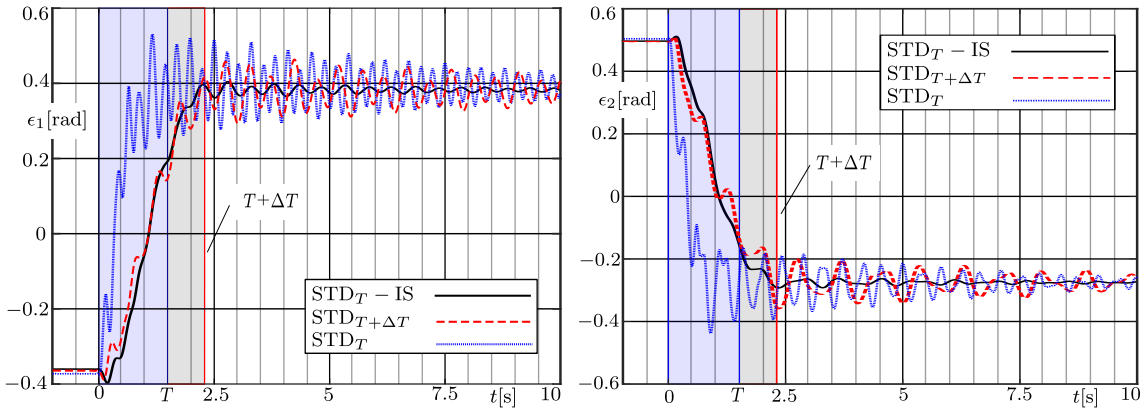
$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{\boldsymbol{\zeta}}_f \\ -\mathbf{M}^{-\perp}_{fP}\left(\mathbf{M}^{\perp}_{cP}\ddot{\boldsymbol{\zeta}}_c + \mathbf{C}^{\perp}_{cP}\dot{\boldsymbol{\zeta}}_c + \mathbf{C}^{\perp}_{fP}\dot{\boldsymbol{\zeta}}_f + \mathbf{f}^{\perp}\right) \end{bmatrix} = \mathbf{f}\Big(\mathbf{x}(t), \boldsymbol{\zeta}_c(t), \dot{\boldsymbol{\zeta}}_c(t), \ddot{\boldsymbol{\zeta}}_c(t)\Big) \tag{5.8}$$

and the assignment of initial rest condition $\mathbf{x}_0$ ($\boldsymbol{\zeta}_{f0}$ is the static equilibrium value for an assigned $\boldsymbol{\zeta}_{c0}$ and $\dot{\boldsymbol{\zeta}}_{f0} = \mathbf{0}_{\lambda}$). Finally, cable lengths can be computed according to the inverse geometric model in (2.15), and fed to low-level motor drivers for manipulator control. While servo-motor angular positions are closed-loop controlled, there is no feedback on the platform pose, and thus its configuration is only feed-forward controlled.

Start and final configurations are selected near the *UACDPR* static workspace edges (see Chapter 3), in order to stress the importance of careful trajectory planning so as to avoid potentially dangerous situations, such as cable loss of tension due to platform large oscillatory motions. $\boldsymbol{\epsilon}$ is expressed by xyz Tait-Bryan angles, since no representation singularities are expected throughout the manipulator static workspace:

$$\boldsymbol{\zeta}_s = [0.36, -0.82, -0.37, -0.35, 0.51, 0.12]^T \text{ [m, rad]}$$
$$\boldsymbol{\zeta}_f = [1.82, 0.55, -0.37, 0.38, -0.25, 0]^T \text{ [m, rad]}$$

Natural oscillation frequencies along the path defined by $\boldsymbol{\zeta}_s$ and $\boldsymbol{\zeta}_f$ vary in the range $[1.19, 2.21]$ Hz (see Fig. 5.2) and are computed by the method described in Section 4.1.



(a) ϵ_1 time evolution

(b) ϵ_2 time evolution

Figure 5.3: Free components of the 4-cable *UACDPR* prototype

Since the ratio between the maximum and minimum frequency is almost 2, a convoluted multi-mode zero-vibration Input Shaper with 3 modes is designed: the 4 pairs (A_i, t_i), $i = 1,\dots,4$ are determined by setting to zero both summations inside the parentheses in Eq. (5.3) for each element in $f = 1.19,\ 1.7,\ 2.21$ Hz (the minimum, mean and maximum frequencies in the range), by considering $t_1 = 0$ s, and imposing $\sum_{i=1}^{k} A_i = 1$; this procedure results in:

$$IS: \quad \begin{array}{c} A_1 = A_4 = 0.1575,\ A_2 = A_3 = 0.3425 \\ t_1 = 0\,\text{s},\ t_1 = 0.294\,\text{s},\ t_3 = 0.588\,\text{s},\ t_4 = 0.882\,\text{s} \end{array} \tag{5.9}$$

with $\Delta T = t_4$. Trapezoidal motion law parameters are selected as $\alpha = 0.2$ and $T = 1.5$ s. Finally, controlled components are selected as $\mathbf{p}$ and ϵ_3. While the choice of $\mathbf{p}$ as part of the controlled coordinates is natural if a positioning task has to be performed, no particular strategy is readily available for the choice of orientation parameters as controlled coordinates. For our demonstrative purpose, any choice is basically good. We decided to choose ϵ_3 just because of its limited variation between the start and final configurations.

Free components ϵ_1 and ϵ_2 time evolution during experiment is shown in Fig. 5.3 as recorded by the Vicon Camera system described in Sec. 4.1. When comparing trajectories with the same total duration, namely STD$_T$ – IS and STD$_{T+\Delta T}$, it is evident that the former allows for smaller amplitude oscillations, which are rapidly damped by unmodelled frictional effects, once the target destination is reached. On the other hand, when comparing unshaped and shaped trajectories, namely STD$_T$ and STD$_T$ – IS, the advantage in employing the latter is even more evident, since the former results in large platform oscillations not only at the final destination, but also during the transition: this fact could easily lead to platform instability and cable loss of tension, thus ultimately robot loss of control.

5.1.2 Robust Oscillation limitation

The natural frequencies of a system are not always precisely known, due to either uncertainty in the dynamic-model parameters or a variation of its internal configuration (both cases occur to a robot in general). Loosely speaking, a trajectory is robust against natural-frequency uncertainty or variation if the amplitude spectrum of $\boldsymbol{\zeta}_c(t) * S(t)$ is

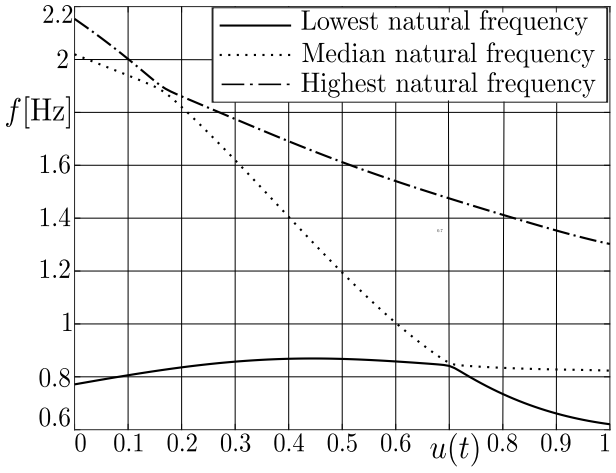


Figure 5.4: *UACDPR* natural frequencies along the specified path

limited over some frequency range. In line-segment paths, for assigned set-points, only $u(t) * S(t)$ is responsible for the location of maximum and minimum values of this amplitude spectrum. If $U(f)$ is the amplitude of the Fourier trasform of $u(t)$, the amplitude spectrum of $u(t) * S(t)$ is $U(f)A_{\%}(f)$.

Limiting the amplitude of $U(f)A_{\%}(f)$ over a frequency range around a given frequency f_0 can be achieved by *Robust IS*, which uses additional impulses, and thus variables (A_i, t_i), to either set $A_{\%}(f_0) = \partial A_{\%}(f_0)/\partial f = ... = \partial^h A_{\%}(f_0)/\partial f^h = 0$ (*ZV-Derivative ZVD IS* [103], Fig. 5.1a), or set $A_{\%}(f) = 0$ in frequencies neighboring f_0 (*Extra-Insensitive EI IS* [105]). In either case, the amplitude of $U(f) * A_{\%}(f)$ is flattened, and thus limited, around f_0. The addition of impulses, though, augments the time delay of the shaper, thus increasing the trajectory duration, and its computational complexity. As an alternative way to obtain similar robustness results on $U(f)A_{\%}(f)$ without the need to modify the *IS*, we propose to use *Dynamic Scaling* (*DS*) of the reference trajectory [106]. An optimal value of T (as well as other trajectory parameters), for a fixed motion law profile (trapezoidal velocity, polynomial, etc..), can be determined by setting $U(f)$, or equivalently $\ddot{U}(f) = -(2\pi f)^2 U(f)$, to zero for the assigned frequency f. This strategy has the additional advantage of determining an upper bound for the total robust-trajectory duration, $T + t_d$. As an example, for a trapezoidal velocity profile, with αT acceleration and deceleration durations ($0 < \alpha \leq 0.5$), α and T can be determined by setting:

$$\ddot{U}(f) = \frac{2}{(1-\alpha)\alpha(\pi T)^2 f} \| \sin[(1-\alpha)\pi f T] \sin[\alpha \pi f T] \| = 0 \tag{5.10}$$

By considering two frequencies f_1 and f_0, such that $f_1 \geq f_0$, and setting the arguments of the two sine functions to π, we obtain $\alpha_{opt} = f_0/(f_1 + f_0)$ and $T_{opt} = (f_1 + f_0)/(f_0 f_1)$, where T_{opt} is strictly decreasing with f_1. It is interesting to notice that, for $f_1 \to f_0$, $\alpha_{opt,s} = 0.5$ and $T_{opt,s} = 2/f_0$, while, for $f_1 \to +\infty$, $\alpha_{opt,l} = 0$ and $T_{opt,l} = 1/f_0$. This in turn means that the optimal time is bounded by the lowest frequency and, for any f, α is always well defined. Moreover, total robust trajectory duration is bounded by: $T_{opt} + t_d \leq T_{opt,s} + t_{d,conv}$.

To verify the effectiveness of the combination of *IS* and *DS*, we consider a linear trajectory of a 6-*DoF* 3-cable *CDPR* with a trapezoidal velocity mo-

tion law. Its model parameters can be found in App.B.2. Natural frequencies along the path between $\boldsymbol{\zeta}_s = [0.793, 1.180, -0.208, 0.058, -0.641, 0.042]$ and $\boldsymbol{\zeta}_f = [-0.826, 1.104, -1.424, -0.041, 0.634, 0.002]$ (units in m and rad) vary as in Fig. 5.4. The equilibrium value of τ_2 in $\boldsymbol{\zeta}_f$ is very small, and potential *EE* oscillation in this configuration could lead to cable slackness. Since the system has 3 frequency spectra, an *IS* with $\mu = 3$ is used, as in [29], but a direct method (with 4 impulses) is employed, and the corresponding frequencies are heuristically selected as the minimum, maximum, and median ones from Fig. 5.4, i.e. $f = 0.621, 1.247, 2.154$Hz. In addition, since $U(f)$ is decreasing with f and only 2 frequencies can be employed for the determination of α_{opt} and T_{opt}, the minimum and median ones are considered for *DS*. These choices lead to:

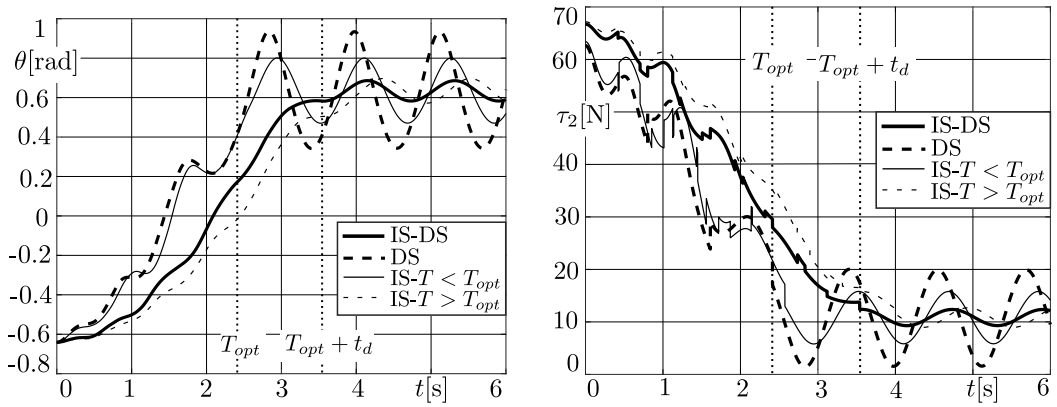
$$IS: \begin{array}{c} A_1 = A_4 = 0.2965,\ A_2 = A_3 = 0.2035 \\ t_1 = 0\,\text{s},\ t_1 = 0.42\,\text{s},\ t_3 = 0.705\,\text{s},\ t_4 = 1.125\,\text{s} \end{array} \qquad DS: \begin{array}{c} \alpha_{opt} = 0.332 \\ T_{opt} = 2.413\,\text{s} \end{array} \tag{5.11}$$

The parameters in Eq. (5.11) are used to compute trajectory in Eq. (5.2), where the controlled coordinates are selected as the position $\mathbf{p}$ of the *EE* reference point ($\mathbf{P} = \mathbf{I}_{6\times6}$). Equation (5.8) is numerically solved with assigned initial conditions $\mathbf{x} = [0.058, -0.641, 0.042, 0, 0, 0]^T$, in order to determine the evolution of the free coordinates along the assigned trajectory. This step allows one to assess oscillations, but also to check for cable slackness. In case these specifications are not satisfactorily met, an intuitive solution may be to robustify *IS*: oscillations are naturally reduced and the increase in time delay bounds cable tensions near their static equilibrium values. Most critical oscillatory variables found *in simulations*[2] are reported in Fig. 5.5 for four trajectories: the first one is planned according to *DS* only, the second one is an *IS* version of the first one (*IS-DS*), the third and fourth ones are *IS* versions of trapezoidal motion laws with $\alpha = 1/3$ (commonly employed value) and $T = 0.6T_{opt} < T_{opt}$ (*IS-*$T < T_{opt}$) and $T = 1.2T_{opt} > T_{opt}$ (*IS-*$T > T_{opt}$), respectively. It is clear that the use of *IS* leads to a more limited oscillatory behavior, and smaller oscillations occur for an increasing T (since the amplitude of $U(f)$ is decreasing with T). However, on the practical side, T should be as small as possible, and *IS* may not lead to satisfactory results if T is too small. Basically, *DS* helps in the choice of the best value of T (and α, for a trapezoidal velocity profile) that optimizes the transition time and limits the oscillatory behavior. Additionally, cable slackness due to oscillations is avoided with *IS-DS* (see Fig. 5.5b, where $\boldsymbol{\tau}$ is computed from Eq. (2.75)). In the end, if critically low oscillations are required, total transition time can always be increased by means of robust *IS*.

5.2 Rest-to-Rest Trajectory Planning

In this Section, we propose a novel trajectory-planning technique that allows the *EE* to track a constrained geometric path *in a specified time*, and allows it to transition *between stable static configuration*: this class of trajectories are called *rest-to-rest*. The design of such a motion is based on the solution of a Boundary Value Problem, formulated as the problem of finding a solution to the *EE* internal dynamics, with constraints on position and velocity at start and end times. Such a trajectory is theoretically possible if model parameters are perfectly known, if the motion law of the system is parametrized specifically, and these parameters are found as the solution of the

[2] Unfortunately, experimental verification was not possible during the time of the PhD thesis due to prototype unavailability and time limitations.



(a) $\theta = \varepsilon_2$ oscillatory behavior (b) τ_2 oscillatory behavior

Figure 5.5: Critical oscillatory behaviors

Boundary Value Problem. The influence of model uncertainties in the execution of a rest-to-rest trajectory was not considered in this thesis and is left as future study.[3]

System theory defines the problem of rest-to-rest trajectory planning for an underactuated mechanical system as a transition problem between stationary setpoints [102]. Such a transition has been proven to be possible [101], in most cases, if the system undergoes an additional *pre-actuation* or *post-actuation* phase, that is, if the system is actuated when $t < 0$, $t > T$ (where T is the transition time) or both. However, this leads to a theoretical impossibility of bringing the system at rest in a predefined time. In addition, a precise path tracking can not be ensured, as the uncontrolled coordinates behavior cannot be predicted, possibly leading the system to instability (i.e. to an oscillatory behavior), thus strongly limiting practical applications.

In [107], a new method was proposed for the trajectory design of Single-Input Single-Output (*SISO*) systems. The same approach was extended to Multi-Input Multi-Output (*MIMO*) systems in [108], and later to systems performing not only single transitions, but also cyclic tasks [109]. These methods ensure that the system is brought to a stationary position in a prescribed time T, but the nominal trajectory of the controlled coordinates undergoes a substantial modification. This is often not desirable (when not dangerous) in industrial applications involving robots, because of possible interference with obstacles.

If the geometric path to track is constrained, a variety of methods may be employed to design the motion law used to track such a path. In the case of a completely or redundantly actuated manipulator the problem of a stationary set point change only requires the solution of a system of linear equations emerging from the fulfilment of some boundary conditions and the necessity of a continuous and differentiable function. Polynomial motion laws are often sufficient to satisfy start- and end-point conditions. An easy way to devise such polynomials is to use so-called transition polynomials [110] of degree $2r+1$:

$$u(t) = \sum_{i=r+1}^{2r+1} a_i \left(\frac{t}{T}\right)^i, \qquad t \in [0, T] \tag{5.12}$$

[3]The content of this Section was published in [45].

where coefficients a_i's do not depend on the task at hand and are given by:

$$a_i = \frac{(-1)^{i-r-1}(2r+1)!}{i \cdot r!(i-r-1)!(2r+1-i)!} \tag{5.13}$$

with (if $r \geq 2$):

$$\sum_{i=r+1}^{2r+1} a_i = 1, \qquad \sum_{i=r+1}^{2r+1} i\,a_i = 0, \qquad \sum_{i=r+1}^{2r+1} i(i-1)\,a_i = 0 \tag{5.14}$$

The index r stands for the maximum order of derivation up to which the continuity of the polynomial is required.

This approach is rarely sufficient for rest-to-rest motions if the system is underactuated, because it does not take into account the internal dynamics in Eq. (2.68). In fact, depending on the inertial effect caused by the geometric path and the chosen motion law, the free coordinates $\boldsymbol{\zeta}_f$ may not reach a stationary condition when the end point is attained.

In order to achieve the desired result, the nonholonomic constraint in Eq. (2.68) must be considered in the planning phase. The stationary conditions on $\boldsymbol{\zeta}_f$ in the start- and end-points can be regarded as boundary conditions (*BCs*) for the differential equation (2.68), thus leading to a *BVP*. This problem has generally no solution when both the path $\boldsymbol{\zeta}_c(u)$ and the motion law $u(t)$ are assigned. On the other hand, if modifications of $u(t)$ are allowed, the problem may admit a solution.

For a rest-to-rest trajectory planning, *BCs* in state form are:

$$\mathbf{x}(0) = \begin{bmatrix} \boldsymbol{\zeta}_f(\boldsymbol{\zeta}_c(0)) \\ \mathbf{0}_{3\times 1} \end{bmatrix} := \mathbf{x}_0, \qquad \mathbf{x}(T) = \begin{bmatrix} \boldsymbol{\zeta}_f(\boldsymbol{\zeta}_c(T)) \\ \mathbf{0}_{3\times 1} \end{bmatrix} := \mathbf{x}_T \tag{5.15}$$

where $[\boldsymbol{\zeta}_c(0)^T \boldsymbol{\zeta}_f(\boldsymbol{\zeta}_c(0))^T]^T$ and $[\boldsymbol{\zeta}_c(T)^T \boldsymbol{\zeta}_f(\boldsymbol{\zeta}_c(T))^T]^T$ are stable equilibrium configurations of the system, that can be obtained as in [15].

Equation (5.8) has dimension 2λ and can only match 2λ out of the 4λ *BCs* established in (5.15). One way to provide a solution to the problem is to consider 2λ additional scalar parameters $\boldsymbol{\kappa} = [\kappa_1, \ldots, \kappa_{2\lambda}]^T$ (called *free parameters*), so that $\boldsymbol{\zeta}_c = \boldsymbol{\zeta}_c(\boldsymbol{\kappa}, t)$. The *BVP* with *free parameters* becomes then:

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{f}\Big(\mathbf{x}(\boldsymbol{\kappa}, t), \boldsymbol{\zeta}_c(\boldsymbol{\kappa}, t), \dot{\boldsymbol{\zeta}}_c(\boldsymbol{\kappa}, t), \ddot{\boldsymbol{\zeta}}_c(\boldsymbol{\kappa}, t)\Big) \\ \mathbf{x}(0) = \mathbf{x}_0, \qquad \mathbf{x}(T) = \mathbf{x}_T \end{cases} \tag{5.16}$$

The solution of (5.16) is a set $\{\boldsymbol{\kappa}, \mathbf{x}(\boldsymbol{\kappa}, t)\}$, where the vector of *free parameters* $\boldsymbol{\kappa} \in \mathbb{R}^{2\lambda}$ is calculated so as the *BC* in Eq. (5.16) are satisfied.

5.2.1 Modification of the Motion Law

In the case of a constrained trajectory geometric path, the motion law is the only element that can undergo a modification, that is, $\boldsymbol{\zeta}_c = \boldsymbol{\zeta}_c(u(\boldsymbol{\kappa}, t))$ [111]. One way to design such a modified motion law u, so that the actuated coordinates can meet the start and end conditions prescribed by the task, is to consider the composition $(u \circ \gamma)(\boldsymbol{\kappa}, t) = u(\gamma(\boldsymbol{\kappa}, t))$, now expressed as:

$$u(\gamma(\boldsymbol{\kappa}, t)) = \sum_{i=r+1}^{2r+1} a_i \gamma^i(\boldsymbol{\kappa}, t) \tag{5.17}$$

$$\gamma(\boldsymbol{\kappa}, 0) = 0, \qquad \gamma(\boldsymbol{\kappa}, T) = 1, \qquad \forall \boldsymbol{\kappa} \in \mathbb{R} \tag{5.18}$$

where a_i is still expressed as in (5.13) and the function $\gamma(\boldsymbol{\kappa}, t)$ is continuous and differentiable up to the second order.

For the purpose of this article, $r = 3$, so that continuity of jerk can be imposed in the start and end positions, and any discontinuity in the cable tensions $\boldsymbol{\tau}$ can be (at least theoretically) avoided. In this way, we try to eliminate a different potential source of residual oscillations.

As an example, $\gamma(\boldsymbol{\kappa}, t)$[4] may be intuitively designed as a polynomial of order $2\lambda+1$:

$$\gamma(\boldsymbol{\kappa}, t) = \alpha t + \sum_{i=2}^{2\lambda+1} \kappa_{i-1} t^i, \qquad \alpha = \frac{1 - \sum_{i=2}^{2\lambda+1} \kappa_{i-1} T^i}{T} \tag{5.19}$$

Accordingly, the time derivative of the actuated coordinates can be expressed as:

$$\dot{\boldsymbol{\zeta}}_c = \frac{\partial \boldsymbol{\zeta}_c}{\partial u} \frac{\partial u}{\partial \gamma} \frac{\partial \gamma}{\partial t} = \boldsymbol{\zeta}_c' u^\star \dot{\gamma} \tag{5.20}$$

where $(\cdot)'$ denotes the partial derivative with respect to u and $(\cdot)^\star$ the partial derivative with respect to γ. The second-order time derivative is hence:

$$\ddot{\boldsymbol{\zeta}}_c = \boldsymbol{\zeta}_c''(u^\star \dot{\gamma})^2 + \boldsymbol{\zeta}_c'(u^\star \ddot{\gamma} + u^{\star\star} \dot{\gamma}^2) \tag{5.21}$$

It should be noted that, as long as conditions (5.18) are satisfied and $r \geq 2$, no other conditions have to be imposed on $\gamma(\boldsymbol{\kappa}, t)$. In fact, $\dot{\boldsymbol{\zeta}}_c(\boldsymbol{\kappa}, T) = 0$ and $\ddot{\boldsymbol{\zeta}}_c(\boldsymbol{\kappa}, T) = 0$ is ensured for any $\boldsymbol{\kappa}$ by $u^\star(\gamma(\boldsymbol{\kappa}, T)) = 0$ and $u^{\star\star}(\gamma(\boldsymbol{\kappa}, T)) = 0$. Other formulations for either $\gamma(\boldsymbol{\kappa}, t)$ or the whole $u(\boldsymbol{\kappa}, t)$ may be employed in order to achieve similar results. From a practical point of view, though, the choice of $\gamma(\boldsymbol{\kappa}, t)$ can affect the convergence rate and speed of the solution algorithm described in Section 5.2.2. In addition, depending on the specific formulation of $\gamma(\boldsymbol{\kappa}, t)$, it may not be easy to determine an initial guess for $\boldsymbol{\kappa}$, which is needed in the numerical solution of Eq. (5.16).

5.2.2 Solution of the *BVP* with Free Parameters

Free parameters $\boldsymbol{\kappa}$ can be found as a consequence of the numerical solution of the *BVP* expressed by Eq. (5.16). A number of algorithms are proposed in the literature and even implemented in commercial softwares, such as the *bvp4c* and *bvp5c* routines available in any MATLAB distribution [112]. These algorithms are finite-difference codes that implement a *collocation formula* [113] and, thus, require a suitable set-up in order to work efficiently and find a solution within a reasonable tolerance. However, even in this case, there is still no guarantee of success.

During our simulation campaign, we were not able to solve problem (5.16) by employing these standard methods, thus we heuristically formulate the problem as a combination of an Initial Value Problem (*IVP*) followed by the solution of a system of nonlinear equations. This approach is similar to a classic iterative *shooting method* [113],

[4] A general formulation for $\gamma(\boldsymbol{\kappa}, t)$ is given by:

$$\gamma(\boldsymbol{\kappa}, t) = \frac{t}{T}\left[\left(1 - \frac{t}{T}\right) h(\boldsymbol{\kappa}, t) + 1\right]$$

where $h(\boldsymbol{\kappa}, t)$ is any continuous and differentiable function up to the second order.

which is a state-of-the-art method for the solution of standard *BVP*s without free parameters: at each iteration of the algorithm, the value of the free parameters is modified instead of the problem initial conditions. For any assigned $\boldsymbol{\kappa}$, let $\mathbf{x}(\boldsymbol{\kappa}, t)$ be the solution of the *IVP* defined by:

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{f}\Big(\mathbf{x}(\boldsymbol{\kappa}, t), \boldsymbol{\zeta}_c(\boldsymbol{\kappa}, t), \dot{\boldsymbol{\zeta}}_c(\boldsymbol{\kappa}, t), \ddot{\boldsymbol{\zeta}}_c(\boldsymbol{\kappa}, t)\Big) \\ \mathbf{x}(0) = \mathbf{x}_0 \end{cases} \tag{5.22}$$

In general, for an arbitrary $\boldsymbol{\kappa}$, $\mathbf{x}(\boldsymbol{\kappa}, t)$ does not meet the end-point condition in Eq. (5.16), namely $\mathbf{x}(T) \neq \mathbf{x}_T$. Consider then the nonlinear equation in the unknown $\boldsymbol{\kappa}$ defined by:

$$\mathbf{F}(\boldsymbol{\kappa}) = \mathbf{x}(\boldsymbol{\kappa}, T) - \mathbf{x}_T = \mathbf{0}_{2\lambda\times 1} \tag{5.23}$$

A solution for equation (5.23) may be found by the following iterative procedure:

0) assign $\boldsymbol{\kappa}_i$ for $i = 0$, e.g. $\boldsymbol{\kappa}_i = \mathbf{0}_{2\lambda\times 1}$, and establish an adequately small tolerance $\epsilon \in \mathbb{R}$;

1) evaluate $\mathbf{x}(\boldsymbol{\kappa}_i, T)$ as the end-point of the solution $\mathbf{x}(\boldsymbol{\kappa}_i, t)$ of the *IVP* (5.22).

2) If $\|\mathbf{F}(\boldsymbol{\kappa}_i)\| \leq \epsilon$, $\{\boldsymbol{\kappa}_i, \mathbf{x}(\boldsymbol{\kappa}_i, t)\}$ is a solution of the *BVP* (5.16), otherwise set $\boldsymbol{\kappa}_{i+1} = \boldsymbol{\kappa}_i + \mathbf{J}_F^{-1}(\boldsymbol{\kappa}_i)\mathbf{F}(\boldsymbol{\kappa}_i)$ and repeat the iteration.

$\mathbf{J}_F(\boldsymbol{\kappa}) = \partial\mathbf{F}/\partial\boldsymbol{\kappa}$ is the Jacobian of Eq. (5.23) with respect to $\boldsymbol{\kappa}$ and it can be approximated by finite differences at every iteration. Finite-difference Jacobian can be efficiently computed by using several parallel threads. However, due to the iterative and approximated nature of the algorithm, the maximum computational time cannot be predicted in advance. Accordingly, this algorithm is not suitable for real-time computation, and has to be employed offline. In our simulations, the algorithm has proven to be sub-linearly convergent in the case it is started from a generic initial guess for $\boldsymbol{\kappa}$.

Once a solution $\{\boldsymbol{\kappa}, \mathbf{x}(\boldsymbol{\kappa}, t)\}$ is found, the trajectory $\boldsymbol{\zeta}_c(\boldsymbol{\kappa}, t)$ may be computed, and the cable total length is found according to Eq. (2.15).

No explicit constraints on cable tensions or motor torques are considered in this work other than positive cable tensions, which are verified during the integration of eq. (5.22)[5]. In addition, it should be noted that the assigned transition time T cannot be arbitrarily low. It is outside the scope of this section to determine an optimal transition time for assigned set-points of a rest-to-rest trajectory. However, in all simulations and experiments conducted, we heuristically determined that a transition time resulting from an average speed of the platform reference point of approximatly 1m/s between assigned set-points always results in a solution of (5.23) with positive tensions in all cables.

5.2.3 Experimental Validation

The trajectory-planning methodology considered in this paper has been implemented in a MATLAB code. For every desired s-th transition between stationary set points, the inputs needed by the trajectory-planner are:

[5]While iteratively solving Eq. (5.23), the IVP (5.22) is integrated at each step, and the values of $\boldsymbol{\tau}(t)$ are calculated by Eq. (2.75) for an assigned $\boldsymbol{\kappa}_i$. In the case a negative value of one tension is determined, the IVP integration is re-initialized with a different value of $\boldsymbol{\kappa}_i$

- the transition time ${}^{s}T$;
- the transition set-points (namely ${}^{s-1}\boldsymbol{\zeta}_c$ and ${}^{s}\boldsymbol{\zeta}_c$);
- the transition geometric path parametrization ${}^{s}\boldsymbol{\zeta}_c(u)$;

After the solution of the problem defined by Eq. (5.16), the outputs of the planning routine are:

- the array of free parameters ${}^{s}\boldsymbol{\kappa}$;
- the controlled and free coordinates, that is, ${}^{s}\boldsymbol{\zeta}_c(t)$ and ${}^{s}\boldsymbol{\zeta}_f(t)$ for $t \in [0, {}^{s}T]$;
- the subsequent cable lengths ${}^{s}l_i(t)(i = 1, \cdots, n)$ for $t \in [0, {}^{s}T]$, determined by (2.15).

In the following, two scenarios will be addressed: i) a 6-*DoF* spatial *UACDPR* actuated by 3 cables whose reference point must track consecutive line segments; ii) the same spatial *UACDPR* as in case (i) whose reference point must track consecutive circular arcs. Robot geometric and inertial parameters can be found in App. B.3.

The platform reference point P will transit and rest in 4 set-points. The stationary (stable) poses of the platform, evaluated as in [15], are:

$$
{}^{0}\boldsymbol{\zeta} = \begin{bmatrix} 1.596 \\ 0.183 \\ -1.300 \\ -0.050 \\ -0.603 \\ -0.575 \end{bmatrix} \qquad
{}^{1}\boldsymbol{\zeta} = \begin{bmatrix} 1.165 \\ 0.211 \\ -0.900 \\ -0.005 \\ -0.210 \\ -0.556 \end{bmatrix} \qquad
{}^{2}\boldsymbol{\zeta} = \begin{bmatrix} 0.587 \\ 0.222 \\ -1.300 \\ 0.009 \\ 0.255 \\ -0.562 \end{bmatrix} \qquad
{}^{3}\boldsymbol{\zeta} = {}^{0}\boldsymbol{\zeta}
$$

and the transition times are:

$$
{}^{1}T = 1.5 \text{ s} \qquad {}^{2}T = 1.5 \text{ s} \qquad {}^{3}T = 2 \text{ s}
$$

Each transition is separated from the next one by a pause of 5 s, so that potential residual oscillations of the platform can be highlighted. *"Rest-to-Rest"* (*RTR*) trajectories, designed according to our approach, are compared with *"Standard"* (*STD*) trajectories, which are defined such as:

$$
\boldsymbol{\kappa} = \mathbf{0}_{2\lambda \times 1}, \qquad \gamma(\boldsymbol{\kappa}, t) = \frac{t}{T} \tag{5.24}
$$

In this case, the end-effector orientation is estimated by forward integration of Eq. (5.22) and the cable lengths by Eq. (2.15). It should be noted that the free coordinates $\boldsymbol{\zeta}_f(t)$ of the *EE*, and thus the lengths of the cables, are generally different for distinct values of $\boldsymbol{\kappa}$. Because of this, the equilibrium poses of the *EE* in the *STD* and *RTR* cases will also be different. In order to be able to compare different subsequent trajectories, *STD* cable lengths were quasi-statically varied during the pause time in order to match the *RTR* ones, and thus have the same start configuration. For the transitions considered in this paper, the difference between them is in the order of some millimiters, so that this procedure does not amplify residual oscillations (as it can be see in Fig. 5.8 and 5.13, where the procedure was simulated).

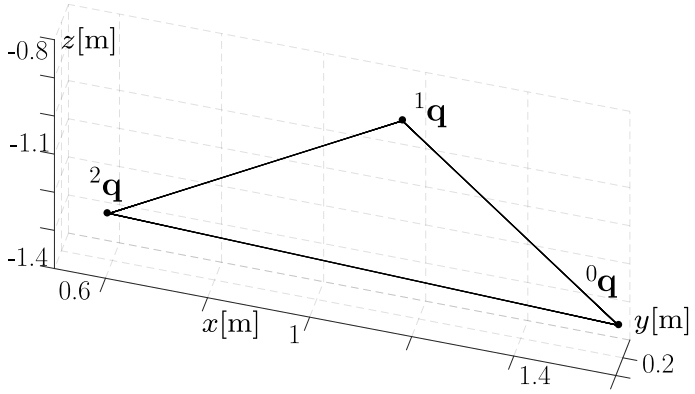
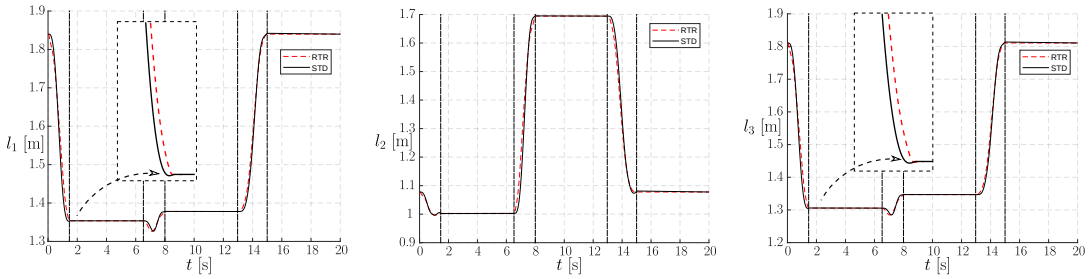
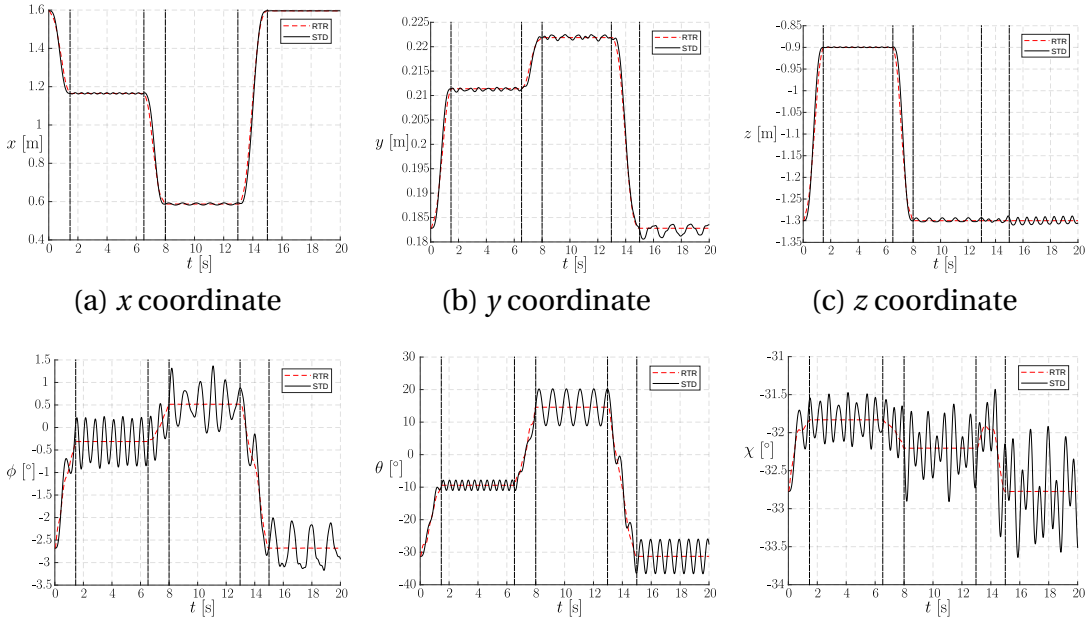


Figure 5.6: Straight line paths.



(a) First cable (b) Second cable (c) Third cable

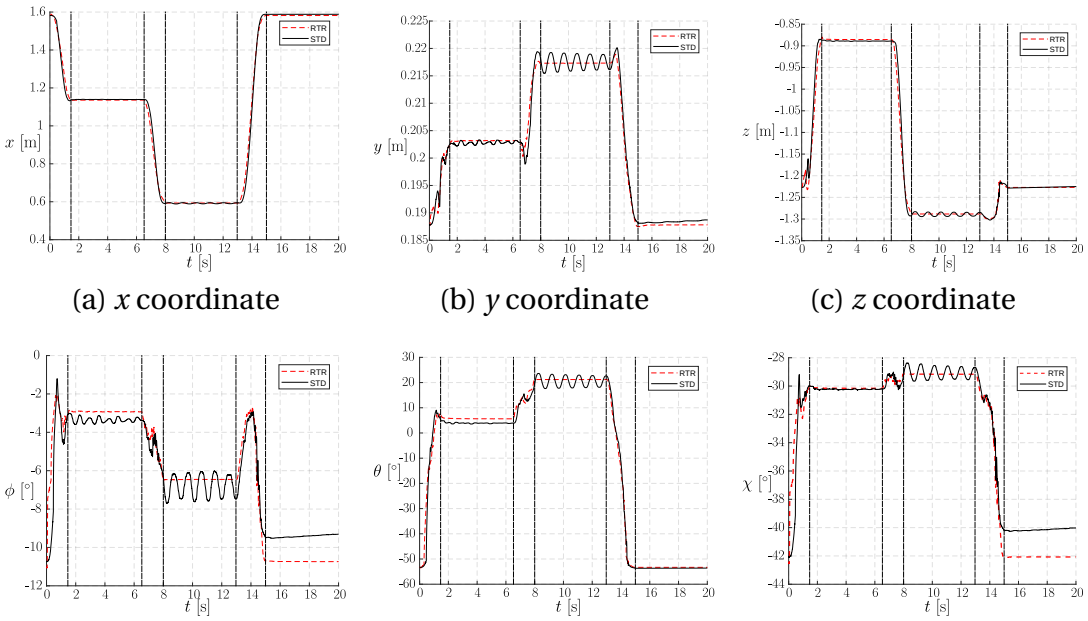
Figure 5.7: Actuator position set-points for *RTR* and *STD* trajectories, in the case of linear paths. Transitions between set-points are delimited by vertical dashed lines



(a) x coordinate (b) y coordinate (c) z coordinate

(d) ϕ Euler angle (e) θ Euler angle (f) χ Euler angle

Figure 5.8: Computed end-effector pose for *RTR* and *STD* trajectories, in the case of linear paths. Transitions between set-points are delimited by vertical dashed lines



(a) x coordinate (b) y coordinate (c) z coordinate

(d) ϕ Euler angle (e) θ Euler angle (f) χ Euler angle

Figure 5.9: Measured end-effector pose for *RTR* and *STD* trajectories, in the case of linear paths. Transitions between set-points are delimited by vertical dashed lines

5.2.3.1 Straight Line Trajectories

In the first example, the set-points are connected by linear paths (Fig. 5.6). The s-th transition is parametrized as:

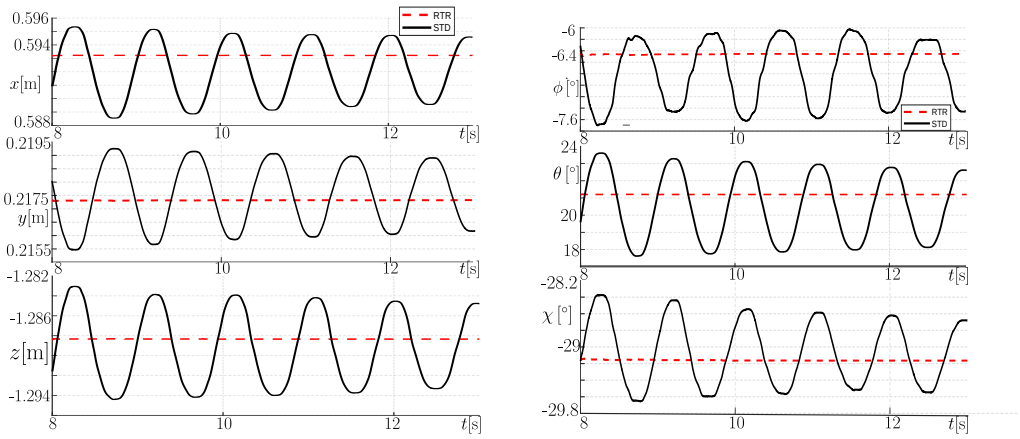
$$ {}^{s}\mathbf{p}(u) = {}^{s-1}\mathbf{p} + ({}^{s}\mathbf{p} - {}^{s-1}\mathbf{p})\, u \tag{5.25} $$

By employing the method outlined in Section 5.2.2, the solution of the *BVP* defined by Eq. (5.16) for $s = 1,2,3$ is found starting from an initial guess $\boldsymbol{\kappa} = \mathbf{0}_{6\times 1}$ in averagely 1.5 min by using a MATLAB implementation installed on a Windows 10 PC, with a 7th generation Intel I7 CPU and 16 Gb of RAM. The results are summarized in Eq. (5.26) and Figs. 5.7 and 5.8:

$$ {}^{1}\boldsymbol{\kappa} = \begin{bmatrix} -14.006 \\ 41.906 \\ -67.565 \\ 60.146 \\ -27.779 \\ 5.195 \end{bmatrix} \quad {}^{2}\boldsymbol{\kappa} = \begin{bmatrix} -12.278 \\ 38.731 \\ -66.101 \\ 61.907 \\ -29.927 \\ 5.827 \end{bmatrix} \quad {}^{3}\boldsymbol{\kappa} = \begin{bmatrix} -4.204 \\ 8.826 \\ -10.157 \\ 6.559 \\ -2.234 \\ 0.312 \end{bmatrix} \tag{5.26} $$

In order to verify the effectiveness of the proposed trajectory planning, no feedback on the platform actual pose was used to stabilize or correct the end-effector position and orientation during experimental testing.

Quantitative results regarding the end-effector pose can be found in Fig. 5.9. Since no external measurement system, such as a laser tracker, was available in our laboratory during the experimental campaign, an indirect approach was used. Angles σ_1, σ_2 and σ_3 were measured by incremental encoders attached to the swivel pulleys' axes. By employing such measurements and the commanded cable lengths, the pose of the platform was estimated making use of a direct kinematic algorithm that employs additional cable orientation measurements [93].



(a) End-effector reference position (b) End-effector Euler angles

Figure 5.10: Enhanced view of most critical measured oscillations during linear paths.

This approach does not allow the platform pose to be inferred with a high precision during a dynamic motion (on the one hand, cables can slightly oscillate in the grooves of their pulleys due to clearance; on the other, pulleys swivel with some lag compared to the theoretical kinematic model, mainly due to friction in the mechanical transmission chains). However, it provides a simple means to effectively compare *RTR* and *STD* trajectories. It is apparent that, though the actuator set-points are very similar for both trajectories, the results in terms of end-effector pose are significantly different. The difference in the global motion of the platform can be explained, in general, by considering that limited difference in the position set-points may be associated with large difference in their higher order derivatives, which play a key role in the dynamics of the system. *STD* trajectories display residual oscillations which are damped over time by dissipative effects and a slight drift in the *EE* coordinates, which may be attributed to cables oscillating into the pulley grooves. On the other hand, *RTR* trajectories results do not show any significant oscillation in the measurements provided by our feedback system (Figs. 5.10a and 5.10b show an enhanced view of the most critical oscillations observed during experiments, for $8\,\mathrm{s} < t < 13\,\mathrm{s}$). We can clearly see that both position and angular *EE* coordinates remain constant after the end of the second transition when *RTR* trajectories are used. The video of the experimentation also displays little to no oscillation left after *RTR* transitions. The small residual swinging of the platform at the end of transitions is due to minor-importance phenomena that are not included in the dynamic model, such as clearance, elasticity, friction, etc.

Also, it can be noted from the planning results (Figs. 5.7a and 5.7c) that, in the first transition, the slope of the cable commanded length in the *RTR* case is steeper compared to the one of the *STD* case. Even though the encoder measurement of our prototype motors show no tracking error, frictional effects in the cable transmission could have led to delays in the actual uncoiling process, thus resulting in a slight modification of the cable effective length during experiment. In addition, a steeper slope means that higher frequency responses of the mechanical system can be excited, such as cables axial vibrations, that were not considered in this work. The second and third transitions required more limited slope changes for cable lengths in the *RTR* case compared to *STD* trajectories, and no detectable oscillations appear in the platform during experiments.

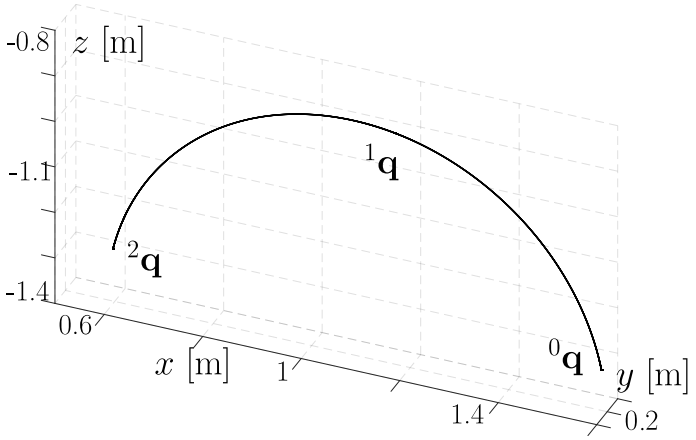
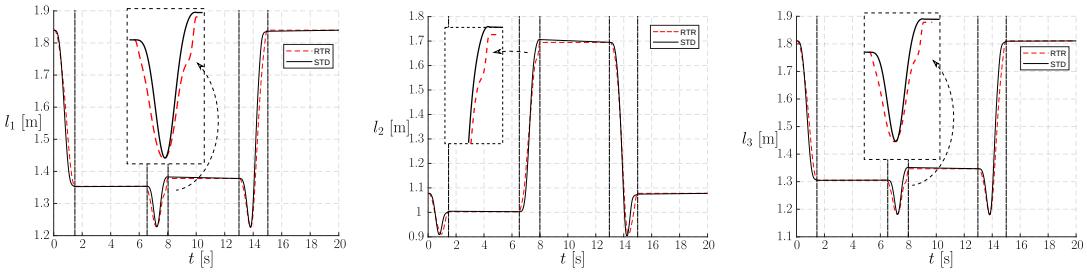
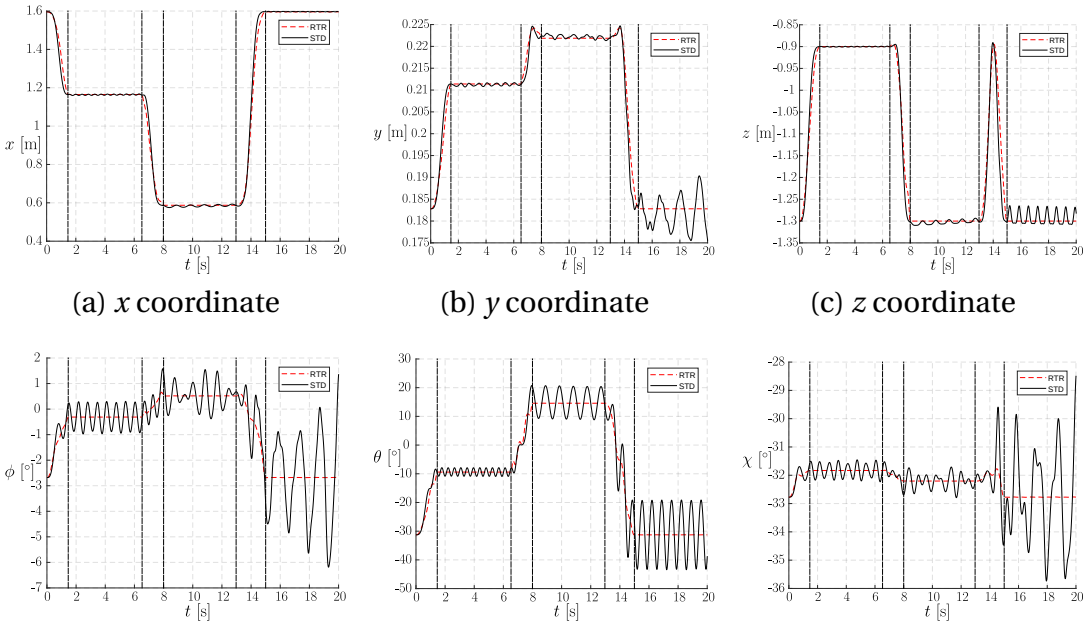


Figure 5.11: Circular geometric paths.



(a) First cable (b) Second cable (c) Third cable

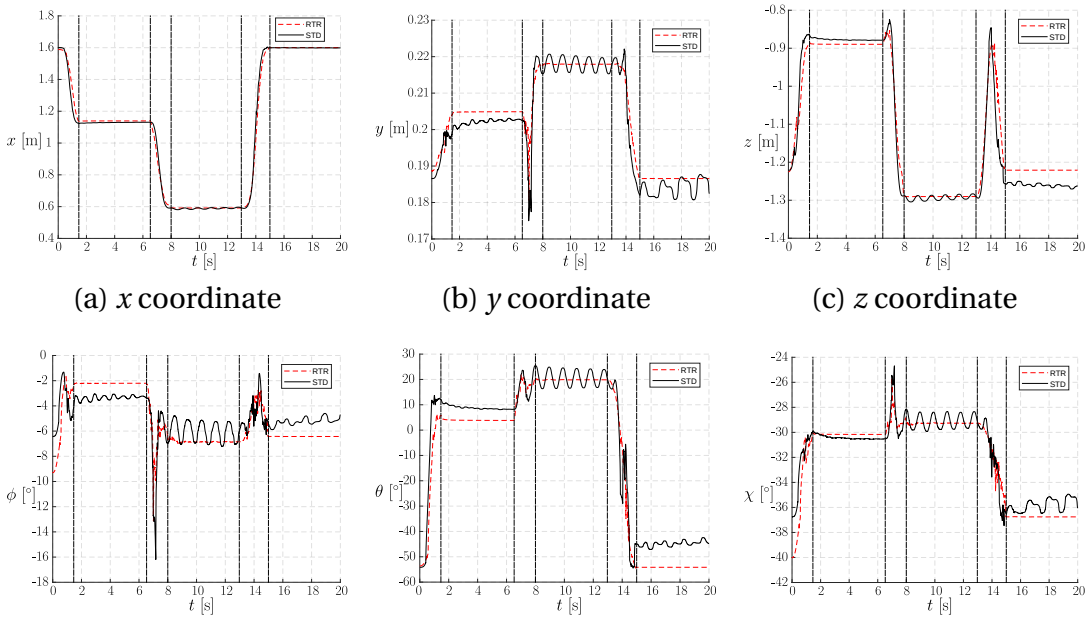
Figure 5.12: Actuator position set-points for *RTR* and *STD* trajectories, in the case of circular paths. Transitions between set-points are delimited by vertical dashed lines



(a) x coordinate (b) y coordinate (c) z coordinate

(d) ϕ Euler angle (e) θ Euler angle (f) χ Euler angle

Figure 5.13: Computed end-effector pose for *RTR* and *STD* trajectories, in the case of circular paths. Transitions between set-points are delimited by vertical dashed lines



(a) x coordinate (b) y coordinate (c) z coordinate

(d) ϕ Euler angle (e) θ Euler angle (f) χ Euler angle

Figure 5.14: Measured end-effector pose for *RTR* and *STD* trajectories, in the case of circular paths. Transitions between set-points are delimited by vertical dashed lines

5.2.3.2 Circular Arc Trajectories

In the second example, circular geometric paths connecting the set-points are considered. The s-th transition is parametrized by the parametric equation of a circular arc passing through 3 points:

$$ {}^{s}\mathbf{p}(u) = \mathbf{c} + r_c\mathbf{x}_c\cos[{}^{s-1}\beta + ({}^{s}\beta - {}^{s-1}\beta)u] + r_c\mathbf{y}_c\sin[{}^{s-1}\beta + ({}^{s}\beta - {}^{s-1}\beta)u] \tag{5.27} $$

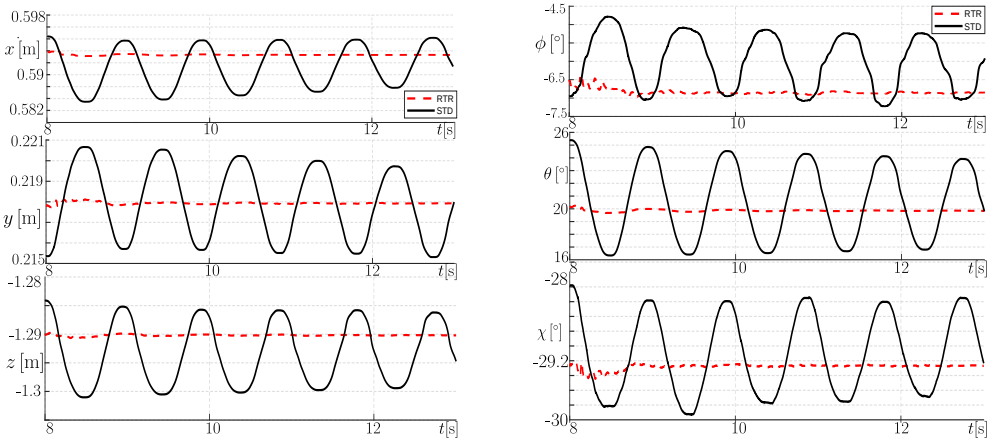
where $\mathbf{c}$ is the center of the circle passing through 3 points ${}^{0}\mathbf{p}$, ${}^{1}\mathbf{p}$ and ${}^{2}\mathbf{p}$, r_c is its radius, $\mathbf{z}_c$ is a unit vector normal to the circle plane, $\mathbf{x}_c = ({}^{0}\mathbf{p} - \mathbf{c})/\|{}^{0}\mathbf{p} - \mathbf{c}\|$, and $\mathbf{y}_c = \mathbf{z}_c \times \mathbf{x}_c$. In addition, angle ${}^{s}\beta$ is defined as:

$$ {}^{s}\beta = \arccos\left[\frac{({}^{s}\mathbf{p} - \mathbf{c}) \cdot ({}^{0}\mathbf{p} - \mathbf{c})}{r_c^2}\right] \text{rad} \tag{5.28} $$

The solution of the *BVP* defined by Eq. (5.16) for $s = 1,2,3$ is found starting from an initial guess $\boldsymbol{\kappa} = \mathbf{0}_{6\times1}$ in averagely 2 min and the results are summarized in Eq. (5.29) and Fig. 5.12 and 5.13:

$$ {}^{1}\boldsymbol{\kappa} = \begin{bmatrix} -4.403 \\ 13.032 \\ -23.118 \\ 22.991 \\ -11.721 \\ 2.386 \end{bmatrix} \quad {}^{2}\boldsymbol{\kappa} = \begin{bmatrix} -18.691 \\ 61.437 \\ -107.706 \\ 102.695 \\ -50.163 \\ 9.812 \end{bmatrix} \quad {}^{3}\boldsymbol{\kappa} = \begin{bmatrix} -3.924 \\ 8.662 \\ -10.559 \\ 7.231 \\ -2.606 \\ 0.384 \end{bmatrix} \tag{5.29} $$

It can be noted in the planning results that in the first transition the slope of the cable commanded lengths is less steep for *RTR* trajectories than for *STD* ones, and the transition occurs smoothly without any oscillation left. The contrary occurs in the second



(a) End-effector reference position

(b) End-effector Euler angles

Figure 5.15: Enhanced view of most critical oscillations during circular paths.

and the third transition. In the second one (Figs. 5.12a to 5.12c), this fact results in a very quick final movement that excites axial vibrations in cables, which are rapidly damped out, but cause a limited oscillatory behavior. Still, the amplitude of these oscillations is not detected by the feedback measurement system, whereas the wide oscillations in the *STD* case are apparent. The third and last transitions, in the *STD* case, display one dangerous effect: the possibility of an unstable behavior of the *EE* due to the lack of constraint (namely, the unactuated *DoFs* evolve with no predetermined boundaries). While transitioning, the platform uncontrolled orientation (see Fig. 5.14) was rapidly changing, almost leading the platform to tip over. In the *RTR* case, instead, the platform slightly oscillates during the transition, and arrives at the rest position with a very limited residual oscillation (Figs. 5.15a and 5.15b show an enhanced view of the most critical oscillations observed during experiments, for $8\,\text{s} < t < 13\,\text{s}$).

It is clear from the results presented in this section that the *EE* may be brought to rest after a transition from one set-point to another only by an accurate trajectory planning that takes into account the internal dynamics of the system.

Chapter 6

Conclusions

Underactuated cable-driven parallel robots (*UACDPRs* in brief) represent a particularly new and not yet fully characterized family of manipulators. This thesis aimed at exploring some of the various issues a robotician faces when dealing with the practical application of this new robotic technology, and at proposing both theoretical and practical solutions.

6.1 Main results

In order to better understand the strengths and the limits of applicability of *UACDPR* technology, the static workspace was characterized by way of a new performance index, called the *maximum tension variation under a unit-norm cable displacement*. This index allowed us to understand which configurations, inside the Wrench Feasible Workspaces (*WFW*) of a *UACDPR*, are more prone to cable becoming slack in presence of actuation errors. Additionally, a novel type of workspace, the *Tension-Error-Insensitive* Workspace (*TEIW*), was defined as the set of poses that a *UACDPR EE* can statically attain even in presence of actuation errors. It was shown that 2-, and 3-cable *UACDPRs TEIWs* are practically coincident with their *WFWs*, whereas the 4-cable robot has a severely limited *TEIW* compared to its *WFW*. 5-cable robots will be studied in the near future.

The end-effector (*EE*) unconstrained motion when cables are not varying their lengths, called *free motion*, was then studied. Small-amplitude free motion was modelled as a harmonic oscillation, and a novel technique for the computation of natural oscillation frequencies for generic *UACDPRs* was proposed. Extensive experimentation on 3 different prototype architectures (2-, 3-, and 4-cable *UACDPRs*) proved our model to be effective in accurately predicting robot oscillation frequencies.

A novel inertial-parameter identification technique for *UACDPR* was then proposed, based on *EE* internal dynamics. This formulation allows for the determination of *EE* dynamic parameters without recurring to any force or torque measurement, which are commonly required by state-of-the-art techniques. Additionally, it was observed that, by using free-motion as identification exciting trajectory, the number of *EE* coordinates to be measured during experiment can be reduced with respect to state-of-the-art methods, without reducing the accuracy of results.

The problem of determining the initial lengths of cables at machine start-up was also addressed. The solution of this problem proved to be crucial for robot safe operation in case only incremental sensors are employed by the *UACDPR*. An automatic

procedure based on a suitable redundant set of incremental sensors, and an extended forward geometrico-static problem, was proposed. Experimental results proved to be satisfactory, especially considering the limited knowledge of the robot geometric parameters and the prototypal nature of the robot used for experiments.

Last, but not least, the problem of trajectory planning of *UACDPRs* was tackled. Two viable solutions were proposed, which are based on different principia and have different strengths. Oscillation-reducing techniques were presented that are computationally fast and can work in real time even if robot parameters are uncertain, but they do not allow for a-priori time-limited transitions between static equilibria. On the contrary, these very strict transitions are possible with a rest-to-rest trajectory planner, but this methodology only allows for off-line planning, in case model parameters are known with high accuracy.

6.2 Open Issues

In the author's opinion, two major problems still need to be thoroughly analyzed, which are pose estimation, and feedback control.

The former problem is particularly involved for underactuated manipulator in general, since actuator deficiency makes it impossible to use forward kinematics to correlate actuator displacements and the *EE* configuration. The use of forward geometrico-static problems limits manipulator performances, since it only allows good tracking performances in quasi-static motions. The consideration of both geometric relationships and *EE* dynamics could provide a solution, but the performance of such an estimation scheme has to be demonstrated, since dynamic models are often simplified in case they need to be computed in real-time. The use of additional proprioceptive sensors, such as encoders on swivel pulleys, may provide a sufficient amount of data for solving an extended forward geometric problem. On the other hand, common pulley modelling relies on rather strict assumptions, which are not completely compatible with applications characterized by non-negligible *EE* dynamics, thus the applicability of such assumptions is to be fully verified.

As far as feedback control is concerned, only simple feedback controllers dedicated to regulate the *EE* pose, while estimating it with approximated methods, are found in the literature. It is the author opinion that an industrial controller of a robotic manipulator should be able to robustly, rapidly, and accurately regulate the robot behaviour in the Cartesian space: this problem is still vastly open.

This thesis spans over several different topics, mostly because during experimental implementation or validation of one solution, a scientific unsolved problem was encountered and needed to be solved. Thus, it is with great honesty that the author admits that he has only scratched the surface of his understanding of *UACDPRs*.

While characterizing the static workspace of *UACDPRs*, we showed that 4-cable *UACDPR* may be intrinsically more difficult to use in real applications,if simple cable-length control strategies are employed. The possibility of employing a hybrid concurrent cable-length / cable-force controller will be addressed in the future, in order to guarantee safe operation inside the whole wrench-feasible workspace of robots with more than 3 cables, while maintaining good tracking performances of a limited set of *EE* coordinates.

Experiments in our laboratory showed us that standard geometric calibration techniques, which only account for robot geometrical equations in the determination of

geometric parameters, are not sufficiently accurate, particularly for those associated with the pulley model. If sensors on pulleys are to be implemented for *EE* pose estimation, dedicated calibration techniques should be devised.

Lastly, the problem of trajectory planning of *UACDPRs* is not yet completely satisfactory for real industrial applications: on the one hand, oscillation-limiting trajectories may not meet strict requirements in terms of transition times, because of the delay imposed by the input-shaping filter, and rest-to-rest trajectories can be limitedly used on systems whose models are not perfectly known. A fusion of these two techniques is currently under investigation, which aims at exploiting the advantages of both techniques.

Appendices

Appendix A

Proofs

This appendix presents some mathematical proofs which were omitted in order to enhance readibility.

A.1 Computation of the swivel angle

Substituting Eq. (2.3) in Eq. (2.5) yields:

$$\left[-\sin(\sigma_i)\,\mathbf{i}_i + \cos(\sigma_i)\,\mathbf{j}_i\right]\cdot\boldsymbol{\varrho}_i = 0 \tag{A.1}$$

and rearranging:

$$\sin(\sigma_i)\left(\mathbf{i}_i\cdot\boldsymbol{\varrho}_i\right) = \cos(\sigma_i)\left(\mathbf{j}_i\cdot\boldsymbol{\varrho}_i\right) \tag{A.2}$$

thus leading to Eq. (2.6):

$$\sigma_i = \operatorname{atan2}\left(\mathbf{j}_i\cdot\boldsymbol{\varrho}_i, \mathbf{i}_i\cdot\boldsymbol{\varrho}_i\right) \tag{A.3}$$

Function atan2$(\cdot)$ is defined for any value of its arguments and allows for the computation of $\sigma_i \in [-\pi,\pi]$.

A.2 Computation of the tangency angle

If we consider:

$$\boldsymbol{\rho}_i = \mathbf{a}_i - \mathbf{b}_i = \mathbf{a}_i - [\mathbf{d}_i + r_i\,(\mathbf{u}_i + \mathbf{n}_i)] = \boldsymbol{\varrho}_i - r_i\,(\mathbf{u}_i + \mathbf{n}_i) \tag{A.4}$$

and substitute it in Eq. (2.9), we obtain:

$$\mathbf{n}_i\cdot\left[\boldsymbol{\varrho}_i - r_i\,(\mathbf{u}_i + \mathbf{n}_i)\right] = \mathbf{n}_i\cdot\boldsymbol{\varrho}_i - r_i\,(\mathbf{n}_i\cdot\mathbf{u}_i + 1) = 0 \tag{A.5}$$

Then, substituting Eq. (2.7) in Eq. (A.5) and rearranging yields:

$$\begin{aligned}\left(\cos\left(\psi_i\right)\mathbf{u}_i + \sin\left(\psi_i\right)\mathbf{k}_i\right)\cdot\boldsymbol{\varrho}_i - r_i\left[\left(\cos\left(\psi_i\right)\mathbf{u}_i + \sin\left(\psi_i\right)\mathbf{k}_i\right)\cdot\mathbf{u}_i + 1\right] = \\ = \left(\mathbf{u}_i\cdot\boldsymbol{\varrho}_i - r_i\right)\cos(\psi_i) + \mathbf{k}_i\cdot\boldsymbol{\varrho}_i\sin(\psi_i) - r_i = \\ = \left(\varrho_{u_i} - r_i\right)\cos(\psi_i) + \varrho_{k_i}\sin(\psi_i) - r_i = 0\end{aligned} \tag{A.6}$$

where $\varrho_{k_i} = \mathbf{k}_i\cdot\boldsymbol{\varrho}_i$ and $\varrho_{u_i} = \mathbf{u}_i\cdot\boldsymbol{\varrho}_i$. If the trigonometric identity:

$$\cos(\psi_i) = \frac{1-t_i^2}{1+t_i^2}, \qquad \sin(\psi_i) = \frac{2\,t_i}{1+t_i^2}, \qquad t_i = \tan(\psi/2) \tag{A.7}$$

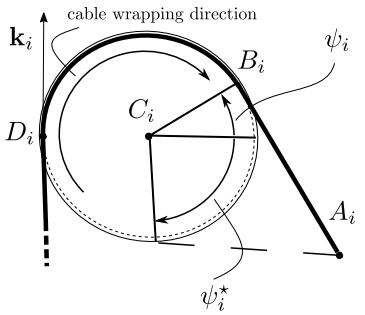


Figure A.1: Geometric interpretation of Eq. (A.9) double solution

is applied to Eq. (A.6), we get:

$$\left(\varrho_{u_i} - r_i\right)\left(1 - t_i^2\right) + 2\varrho_{k_i} t_i - r_i\left(1 + t_i^2\right) = -\varrho_{u_i} t_i^2 + 2\varrho_{k_i} t_i + \varrho_{u_i} - 2r_i = 0 \tag{A.8}$$

which can be straightforwardly solved for t_i as:

$$t_i = \tan(\psi/2) = \frac{\varrho_{k_i}}{\varrho_{u_i}} \pm \sqrt{\left(\frac{\varrho_{k_i}}{\varrho_{u_i}}\right)^2 + 1 - \frac{2r_i}{\varrho_{u_i}}} \tag{A.9}$$

According to the cable being clockwise wrapped onto the pulley, the only physical solution for t_i is the one with the positive sign in front of the square root (see Fig. A.1, where the alternative solution is marked as $\psi_i^\star$). Finally Eq. (2.10) is obtained by inverting the tan function and rearranging as:

$$\psi_i = 2\,\mathrm{atan}\left[\frac{\varrho_{k_i}}{\varrho_{u_i}} + \sqrt{\left(\frac{\varrho_{k_i}}{\varrho_{u_i}}\right)^2 + 1 - \frac{2r_i}{\varrho_{u_i}}}\right] \tag{A.10}$$

A.3 Computation of the swivel angle time derivative

The time derivative of Eq. (2.5) yields:

$$\dot{\mathbf{w}}_i \cdot \boldsymbol{\varrho}_i + \mathbf{w}_i \cdot \dot{\boldsymbol{\varrho}}_i = 0 \tag{A.11}$$

Since $\dot{\boldsymbol{\varrho}}_i = \dot{\mathbf{a}}_i$, we may compute the velocity of point A_i and the rate of change of $\mathbf{w}_i$ from Eqs. (2.2) and (2.3) as:

$$\dot{\mathbf{a}}_i = \dot{\mathbf{p}} + \boldsymbol{\omega} \times \mathbf{a}_i', \qquad \dot{\mathbf{w}}_i = -\dot{\sigma}_i \mathbf{u}_i \tag{A.12}$$

Substituting in Eq. (A.11) and rearranging yields:

$$\mathbf{u}_i \cdot \boldsymbol{\varrho}_i \dot{\sigma}_i = \varrho_{u_i} \dot{\sigma}_i = \mathbf{w}_i \cdot \dot{\mathbf{p}} + \mathbf{w}_i \cdot \boldsymbol{\omega} \times \dot{\mathbf{a}}_i' = \dot{\mathbf{p}} \cdot \mathbf{w}_i + \boldsymbol{\omega} \cdot \dot{\mathbf{a}}_i' \times \mathbf{w}_i \tag{A.13}$$

thus finally:

$$\dot{\sigma}_i = \frac{1}{\varrho_{u_i}}\left(\dot{\mathbf{p}} \cdot \mathbf{w}_i + \boldsymbol{\omega} \cdot \dot{\mathbf{a}}_i' \times \mathbf{w}_i\right) = \boldsymbol{\xi}_{\sigma_i} \cdot \mathbf{v} \tag{A.14}$$

A.4 Computation of the tangency angle time derivative

The time derivative of Eq. (2.9) yields:

$$\dot{\mathbf{n}}_i \cdot \boldsymbol{\rho}_i + \mathbf{n}_i \cdot \dot{\boldsymbol{\rho}}_i = 0 \tag{A.15}$$

Its terms can be computed by differentiating w.r.t. time Eqs. (2.11), :

$$\dot{\boldsymbol{\rho}}_i = \dot{\mathbf{a}}_i - r_i\,(\dot{\mathbf{u}}_i + \dot{\mathbf{n}}_i)\,, \qquad \dot{\mathbf{u}}_i = \dot{\sigma}_i \mathbf{w}_i, \qquad \dot{\mathbf{n}}_i = \cos(\psi_i)\dot{\sigma}_i \mathbf{w}_i - \dot{\psi}_i \mathbf{t}_i \tag{A.16}$$

Substituting in Eq. (A.15) and rearranging yields:

$$\mathbf{t}_i \cdot \boldsymbol{\rho}_i \dot{\psi}_i = \|\boldsymbol{\rho}_i\| \dot{\psi}_i = \mathbf{n}_i \cdot \dot{\mathbf{p}} + \mathbf{n}_i \cdot \boldsymbol{\omega} \times \dot{\mathbf{a}}'_i = \dot{\mathbf{p}} \cdot \mathbf{n}_i + \boldsymbol{\omega} \cdot \dot{\mathbf{a}}'_i \times \mathbf{n}_i \tag{A.17}$$

since $\mathbf{w}_i \cdot \mathbf{n}_i = \mathbf{t}_i \cdot \mathbf{n}_i = \boldsymbol{\rho}_i \cdot \mathbf{n}_i = 0$. Finally:

$$\dot{\psi}_i = \frac{1}{\|\boldsymbol{\rho}_i\|} \left(\dot{\mathbf{p}} \cdot \mathbf{n}_i + \boldsymbol{\omega} \cdot \dot{\mathbf{a}}'_i \times \mathbf{n}_i\right) = \boldsymbol{\xi}_{\psi_i} \cdot \mathbf{v} \tag{A.18}$$

A.5 Computation of the cable length time derivative

The time derivative of Eq. (2.14) yields:

$$2\dot{\boldsymbol{\rho}}_i \cdot \boldsymbol{\rho}_i - 2\|\boldsymbol{\rho}_i\| \frac{\partial \|\boldsymbol{\rho}_i\|}{\partial t} = 0 \tag{A.19}$$

The time derivative of $\|\boldsymbol{\rho}_i\|$ is computed from Eq. (2.13) as:

$$\frac{\partial \|\boldsymbol{\rho}_i\|}{\partial t} = \dot{l}_i + r_i \dot{\psi}_i \tag{A.20}$$

Substituting Eqs. (2.12), the left-hand side of Eq. (A.16) and (A.20) in Eq. (A.19) yields:

$$\|\boldsymbol{\rho}_i\| \dot{\mathbf{a}}_i \cdot \mathbf{t}_i - r_i \|\boldsymbol{\rho}_i\| \left[\left(\cos(\psi_i) + 1\right) \dot{\sigma}_i \mathbf{w}_i - \dot{\psi}_i \mathbf{t}_i\right] \cdot \mathbf{t}_i - \|\boldsymbol{\rho}_i\| (\dot{l}_i + r_i \dot{\psi}_i) = 0 \tag{A.21}$$

Simplifying and rearranging Eq. (A.21) finally gives:

$$\dot{l}_i = \left(\dot{\mathbf{p}} \cdot \mathbf{t}_i + \boldsymbol{\omega} \cdot \dot{\mathbf{a}}'_i \times \mathbf{t}_i\right) = \boldsymbol{\xi}_{l_i} \cdot \mathbf{v} \tag{A.22}$$

A.6 Swivel angle second order time derivative

The time derivative of Eq. (A.3) yields:

$$\left[\dot{\mathbf{u}}_i \cdot \boldsymbol{\varrho}_i + \mathbf{u}_i \cdot \dot{\boldsymbol{\varrho}}_i\right] \dot{\sigma}_i + \varrho_{u_i} \ddot{\sigma}_i = \dot{\mathbf{w}}_i \cdot \dot{\mathbf{a}}_i + \mathbf{w}_i \cdot \ddot{\mathbf{a}}_i \tag{A.23}$$

and accounting for Eqs. (2.5), (A.12) and:

$$\ddot{\mathbf{a}}_i = \ddot{\mathbf{p}} + \boldsymbol{\alpha} \times \mathbf{a}'_i + \boldsymbol{\omega} \times \left(\boldsymbol{\omega} \times \mathbf{a}'_i\right) \tag{A.24}$$

one has:

$$\ddot{\sigma}_i = -\frac{2}{\varrho_{u_i}} (\boldsymbol{\xi}_{u_i} \cdot \mathbf{v})(\boldsymbol{\xi}_{\sigma_i} \cdot \mathbf{v}) + \boldsymbol{\xi}_{\sigma_i} \cdot \dot{\mathbf{v}} + \frac{\mathbf{w}_i}{\varrho_{u_i}} \cdot \boldsymbol{\omega} \times \left(\boldsymbol{\omega} \times \mathbf{a}'_i\right), \qquad \boldsymbol{\xi}_{u_i} = \begin{bmatrix} \mathbf{u}_i \\ \mathbf{a}'_i \times \mathbf{u}_i \end{bmatrix} \tag{A.25}$$

Thanks to the Jacobi identity of the vector product, one has:

$$\mathbf{w}_i \cdot \boldsymbol{\omega} \times \left(\boldsymbol{\omega} \times \mathbf{a}_i'\right) = \boldsymbol{\omega} \cdot \mathbf{a}_i' \times (\mathbf{w}_i \times \boldsymbol{\omega}) \tag{A.26}$$

and Eq. (A.25) can be rewritten in matrix form as:

$$\ddot{\sigma}_i = \mathbf{v}^T \boldsymbol{\xi}_{\sigma_i}' \mathbf{v} + \boldsymbol{\xi}_{\sigma_i}^T \dot{\mathbf{v}} \tag{A.27}$$

where:

$$\boldsymbol{\xi}_{\sigma_i}' = \frac{1}{\varrho_{u_i}} \left(-2\boldsymbol{\xi}_{u_i} \boldsymbol{\xi}_{\sigma_i}^T + \mathbf{A}_{w_i}\right), \qquad \mathbf{A}_{w_i} = \begin{bmatrix} \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \tilde{\mathbf{a}}_i' \tilde{\mathbf{w}}_i \end{bmatrix} \tag{A.28}$$

It should be noted that:

$$\ddot{\sigma}_i = \dot{\boldsymbol{\xi}}_{\sigma_i} \cdot \mathbf{v} + \boldsymbol{\xi}_{\sigma_i} \cdot \dot{\mathbf{v}}, \qquad \dot{\boldsymbol{\xi}}_{\sigma_i} = \boldsymbol{\xi}_{\sigma_i}' \mathbf{v} \tag{A.29}$$

A.7 Tangency angle second order time derivative

The time derivative of Eq. (A.17) yields:

$$\frac{\partial \|\boldsymbol{\rho}_i\|}{\partial t} \dot{\psi}_i + \|\boldsymbol{\rho}_i\| \ddot{\psi}_i = \dot{\mathbf{n}}_i \cdot \dot{\mathbf{a}}_i + \mathbf{n}_i \cdot \ddot{\mathbf{a}}_i \tag{A.30}$$

If one accounts for Eqs. (A.16) and (A.20), one has:

$$\ddot{\psi}_i = \frac{1}{\|\boldsymbol{\rho}_i\|} \left(\varrho_{u_i} \cos(\psi_i) \dot{\sigma_i}^2 - r_i \dot{\psi}_i^2 - 2\dot{l}_i \dot{\psi}_i + \mathbf{n}_i \cdot \boldsymbol{\omega} \times \left(\boldsymbol{\omega} \times \mathbf{a}_i'\right)\right) + \boldsymbol{\xi}_{\psi_i} \cdot \dot{\mathbf{v}} \tag{A.31}$$

which can be written in matrix form, by using the Jacobi identity and Eqs. (A.14), (A.18) and (A.22), as:

$$\ddot{\psi}_i = \mathbf{v}^T \boldsymbol{\xi}_{\psi_i}' \mathbf{v} + \boldsymbol{\xi}_{\psi_i}^T \dot{\mathbf{v}} \tag{A.32}$$

where:

$$\boldsymbol{\xi}_{\psi_i}' = \frac{1}{\|\boldsymbol{\rho}_i\|} \left(\varrho_{u_i} \cos(\psi_i) \boldsymbol{\xi}_{\sigma_i} \boldsymbol{\xi}_{\sigma_i}^T - r_i \boldsymbol{\xi}_{\psi_i} \boldsymbol{\xi}_{\psi_i}^T - 2\boldsymbol{\xi}_{l_i} \boldsymbol{\xi}_{\psi_i}^T + \mathbf{A}_{n_i}\right), \quad \mathbf{A}_{n_i} = \begin{bmatrix} \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \tilde{\mathbf{a}}_i' \tilde{\mathbf{n}}_i \end{bmatrix} \tag{A.33}$$

It should be noted that:

$$\ddot{\psi}_i = \dot{\boldsymbol{\xi}}_{\psi_i} \cdot \mathbf{v} + \boldsymbol{\xi}_{\psi_i} \cdot \dot{\mathbf{v}}, \qquad \dot{\boldsymbol{\xi}}_{\psi_i} = \boldsymbol{\xi}_{\psi_i}' \mathbf{v} \tag{A.34}$$

A.8 Cable length second order time derivative

The time derivative of Eq. (A.22) yields:

$$\ddot{l}_i = \dot{\mathbf{t}}_i \cdot \dot{\mathbf{a}}_i + \mathbf{t}_i \cdot \ddot{\mathbf{a}}_i \tag{A.35}$$

where $\dot{\mathbf{t}}_i$ is evaluated as the time derivative of Eq. (2.8) as:

$$\dot{\mathbf{t}}_i = \sin(\psi_i) \dot{\sigma}_i \mathbf{w}_i + \dot{\psi}_i \mathbf{n}_i \tag{A.36}$$

Then, according to Eqs. (A.14) and (A.18), one has;

$$\ddot{l}_i = \sin(\psi_i) \varrho_{u_i} \dot{\sigma}_i^2 + \|\boldsymbol{\rho}_i\| \dot{\psi}_i^2 + \mathbf{t}_i \cdot \boldsymbol{\omega} \times \left(\boldsymbol{\omega} \times \mathbf{a}_i'\right) + \boldsymbol{\xi}_{l_i} \cdot \dot{\mathbf{v}} \tag{A.37}$$

which can be written in matrix form, by using the Jacobi identity and Eqs. (A.14) and (A.18) as:

$$\ddot{l}_i = \mathbf{v}^T \boldsymbol{\xi}'_{l_i} \mathbf{v} + \boldsymbol{\xi}_{l_i}^T \dot{\mathbf{v}} \tag{A.38}$$

where:

$$\boldsymbol{\xi}'_{l_i} = \sin(\psi_i)\varrho_{u_i}\boldsymbol{\xi}_{\sigma_i}\boldsymbol{\xi}_{\sigma_i}^T + \|\boldsymbol{\rho}_i\|\boldsymbol{\xi}_{\psi_i}\boldsymbol{\xi}_{\psi_i}^T + \mathbf{A}_{t_i}, \qquad \mathbf{A}_{t_i} = \begin{bmatrix} \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \tilde{\mathbf{a}}'_i\tilde{\mathbf{t}}_i \end{bmatrix} \tag{A.39}$$

It should be noted that:

$$\ddot{l}_i = \dot{\boldsymbol{\xi}}_{l_i} \cdot \mathbf{v} + \boldsymbol{\xi}_{l_i} \cdot \dot{\mathbf{v}}, \qquad \dot{\boldsymbol{\xi}}_{l_i} = \boldsymbol{\xi}'_{l_i}\mathbf{v} \tag{A.40}$$

A.9 Tangent vector first order time derivative

The time derivative of Eq. (2.8) yields:

$$\dot{\mathbf{t}}_i = \left(\cos\left(\psi_i\right)\mathbf{u}_i + \sin\left(\psi_i\right)\mathbf{k}_i\right)\dot{\psi}_i + \sin\left(\psi_i\right)\dot{\mathbf{u}}_i \tag{A.41}$$

If we substitute the results of Eqs. (2.7) and (A.16) in Eq. (A.41), we get:

$$\dot{\mathbf{t}}_i = \mathbf{n}_i\dot{\psi}_i + \sin\left(\psi_i\right)\mathbf{w}_i\dot{\sigma}_i \tag{A.42}$$

Finally, substituting Eqs. (A.3) and (A.4) in Eq. (A.42) yields:

$$\dot{\mathbf{t}}_i = \left(\mathbf{n}_i\boldsymbol{\xi}_{\psi_i}^T + \sin\left(\psi_i\right)\mathbf{w}_i\boldsymbol{\xi}_{\sigma_i}^T\right)\mathbf{v} \tag{A.43}$$

Then, if we consider the definition of Eq. (2.17):

$$\dot{\mathbf{t}}_i = \left(\mathbf{n}_i\boldsymbol{\xi}_{\psi_i}^T + \sin\left(\psi_i\right)\mathbf{w}_i\boldsymbol{\xi}_{\sigma_i}^T\right)\mathbf{D}\dot{\boldsymbol{\zeta}} \tag{A.44}$$

we can also deduce:

$$\frac{\partial\mathbf{t}_i}{\partial\boldsymbol{\zeta}} = \left(\mathbf{n}_i\boldsymbol{\xi}_{\psi_i}^T + \sin\left(\psi_i\right)\mathbf{w}_i\boldsymbol{\xi}_{\sigma_i}^T\right)\mathbf{D} \tag{A.45}$$

which can be written in matrix form as:

$$\frac{\partial\mathbf{t}_i}{\partial\boldsymbol{\zeta}} = \begin{bmatrix}\mathbf{T}_i & -\mathbf{T}_i\tilde{\mathbf{a}}'_i\end{bmatrix}\mathbf{D}, \qquad \mathbf{T}_i = \frac{\sin\psi_i\mathbf{w}_i\mathbf{w}_i^T}{\varrho_{u_i}} + \frac{\mathbf{n}_i\mathbf{n}_i^T}{\|\boldsymbol{\rho}_i\|} \tag{A.46}$$

A.10 Free-Motion Cable Tension Variation

In order to determine the free-motion cable tension variation, Eq.(2.81), which is here reported for the sake of convenience:

$$\mathbf{M}_f^{\|}\ddot{\boldsymbol{\zeta}}_f + \mathbf{C}_f^{\|}\dot{\boldsymbol{\zeta}}_f + \mathbf{f}^{\|} = \boldsymbol{\tau} \tag{A.47}$$

has to be linearized about an equilibrium configuration.

To this end, if the left-hand side of Eq. (A.47) is denoted as $\mathbf{h}\left(\boldsymbol{\zeta}_f, \dot{\boldsymbol{\zeta}}_f, \ddot{\boldsymbol{\zeta}}_f\right)$, the Taylor expansion about an equilibrium configuration ($\boldsymbol{\zeta}_f = \boldsymbol{\zeta}_{f0}$, $\dot{\boldsymbol{\zeta}}_f = \mathbf{0}_{6\times1}$, $\ddot{\boldsymbol{\zeta}}_f = \mathbf{0}_{6\times1}$) truncated at the first order yields:

$$\begin{aligned}\boldsymbol{\tau} = \mathbf{h}\left(\boldsymbol{\zeta}_f, \dot{\boldsymbol{\zeta}}_f, \ddot{\boldsymbol{\zeta}}_f\right) \simeq \mathbf{h}(\boldsymbol{\zeta}_{f0}, \mathbf{0}, \mathbf{0}) + \\ + \left.\frac{\partial\mathbf{h}}{\partial\ddot{\boldsymbol{\zeta}}_f}\right|_{(\boldsymbol{\zeta}_{f0},\mathbf{0},\mathbf{0})}\ddot{\boldsymbol{\zeta}}_f + \left.\frac{\partial\mathbf{h}}{\partial\dot{\boldsymbol{\zeta}}_f}\right|_{(\boldsymbol{\zeta}_{f0},\mathbf{0},\mathbf{0})}\dot{\boldsymbol{\zeta}}_f + \left.\frac{\partial\mathbf{h}}{\partial\boldsymbol{\zeta}_f}\right|_{(\boldsymbol{\zeta}_{f0},\mathbf{0},\mathbf{0})}\left(\boldsymbol{\zeta}_f - \boldsymbol{\zeta}_{f0}\right)\end{aligned} \tag{A.48}$$

At equilibrium, clearly $\mathbf{h}\left(\boldsymbol{\zeta}_{f0},\mathbf{0},\mathbf{0}\right)=\boldsymbol{\tau}_0$. The partial derivatives are readily obtained as:

$$\left.\frac{\partial \mathbf{h}}{\partial \ddot{\boldsymbol{\zeta}}_f}\right|_{(\boldsymbol{\zeta}_{f0},\mathbf{0},\mathbf{0})}=\left.\mathbf{M}_f^{\parallel}\right|_{(\boldsymbol{\zeta}_{f0},\mathbf{0},\mathbf{0})}=\mathbf{M}_{f0}^{\parallel} \tag{A.49}$$

$$\left.\frac{\partial \mathbf{h}}{\partial \dot{\boldsymbol{\zeta}}_f}\right|_{(\boldsymbol{\zeta}_{f0},\mathbf{0},\mathbf{0})}=\left.\left(\mathbf{C}_f^{\parallel}+\frac{\partial \mathbf{C}_f^{\parallel}}{\partial \dot{\boldsymbol{\zeta}}_f}\dot{\boldsymbol{\zeta}}_f\right)\right|_{(\boldsymbol{\zeta}_{f0},\mathbf{0},\mathbf{0})}=\mathbf{0}_{\lambda\times\lambda} \tag{A.50}$$

$$\begin{aligned}\left.\frac{\partial \mathbf{h}}{\partial \boldsymbol{\zeta}_f}\right|_{(\boldsymbol{\zeta}_{f0},\mathbf{0},\mathbf{0})}&=\left.\left(\frac{\partial \mathbf{M}_f^{\parallel}}{\partial \boldsymbol{\zeta}_f}\ddot{\boldsymbol{\zeta}}_f+\frac{\partial \mathbf{C}_f^{\parallel}}{\partial \boldsymbol{\zeta}_f}\dot{\boldsymbol{\zeta}}_f+\frac{\partial \mathbf{f}^{\parallel}}{\partial \boldsymbol{\zeta}_f}\right)\right|_{(,\mathbf{0},\mathbf{0})}=\\&=\left.\frac{\partial \mathbf{f}^{\parallel}}{\partial \boldsymbol{\zeta}_f}\right|_{(\boldsymbol{\zeta}_{f0},\mathbf{0},\mathbf{0})}=\left.\mathbf{K}_f^{\parallel}\right|_{(\boldsymbol{\zeta}_{f0},\mathbf{0},\mathbf{0})}=\mathbf{K}_{f0}^{\parallel}\end{aligned} \tag{A.51}$$

where we have taken advantages that many elements vanishing in Eqs. from (A.49) to (A.51) are linearly dependent on $\dot{\boldsymbol{\zeta}}_f$ and $\ddot{\boldsymbol{\zeta}}_f$. All quantities at the far right-hand sides from (A.49) to (A.51) are computed in the equilibrium configuration $(\boldsymbol{\zeta}_{f0},\mathbf{0},\mathbf{0})$ and thus denoted by the subscript 0. Matrix $\mathbf{M}_f^{\parallel}$ is given in Eq. (2.79) and is reported below for the sake of convenience:

$$\mathbf{M}_f^{\parallel}=-{\boldsymbol{\Xi}_l^{\parallel}}^T\mathbf{M}\boldsymbol{\Xi}_l^{\perp}\in\mathbb{R}^{n\times\lambda} \tag{A.52}$$

Matrix $\mathbf{K}_f^{\parallel}$ can be derived according to the results of Sec. 2.4.2:

$$\mathbf{K}_f^{\parallel}=\frac{\partial \mathbf{f}^{\parallel}}{\partial \boldsymbol{\zeta}_f}=\frac{\partial \mathbf{f}^{\parallel}}{\partial \boldsymbol{\zeta}}\frac{\partial \boldsymbol{\zeta}}{\partial \boldsymbol{\zeta}_f} \tag{A.53}$$

and considering Eqs.(2.114) and (2.115):

$$\mathbf{K}_f^{\parallel}=-{\boldsymbol{\Xi}_l^{\parallel}}^T\left[\left(\mathbf{K}+\mathbf{E}\right)\mathbf{D}-\mathbf{F}\right]\mathbf{J}_l^{\perp} \tag{A.54}$$

Finally, Eq. (A.48) can be rewritten as:

$$\Delta\boldsymbol{\tau}_0=\boldsymbol{\tau}-\boldsymbol{\tau}_0=\mathbf{M}_{f0}^{\parallel}\ddot{\boldsymbol{\zeta}}_f+\mathbf{K}_{f0}^{\parallel}\left(\boldsymbol{\zeta}_f-\boldsymbol{\zeta}_{f0}\right)=\mathbf{M}_{f0}^{\parallel}\Delta\ddot{\boldsymbol{\zeta}}_{f0}+\mathbf{K}_{f0}^{\parallel}\Delta\boldsymbol{\zeta}_{f0} \tag{A.55}$$

where $\Delta\ddot{\boldsymbol{\zeta}}_{f0}=\ddot{\boldsymbol{\zeta}}_f-\mathbf{0}_{\lambda\times1}$, $\Delta\boldsymbol{\zeta}_{f0}=\boldsymbol{\zeta}_f-\boldsymbol{\zeta}_{f0}$, and $\Delta\boldsymbol{\tau}_0=\boldsymbol{\tau}-\boldsymbol{\tau}_0$.

Appendix B

Geometrical and Inertial Data of *UACDPRs*

This appendix collects all the geometrical and inertial data of the *UACDPRs* used in this thesis. These data are primarily derived by the mechanical evolution of the prototype of the university available at Bologna and were obtained either by Computer-Aided-Design models or by calibration and identification techniques.

B.1 Data for Workspace Computation

Geometrical and inertial properties of the prototype used for workspace analyses (Fig. B.1), defined in Sections 2.1 and 2.3, are summarized in Tables B.1 and B.2, where $\mathbf{i} = [1;0;0]^T$, $\mathbf{j} = [0;1;0]^T$, and $\mathbf{k} = [0;0;1]^T$, and the only external load applied to the robot *EE* is gravity, thus $\mathbf{e}' = \mathbf{s}'$, $\boldsymbol{\phi} = -mg\mathbf{k}$ and $\boldsymbol{\mu} = \mathbf{0}_{3\times3}$. The coordinates of $\mathbf{a}'_i$, $\mathbf{s}'$ and $\mathbf{I}_G$ are constant in the *EE* frame, and denoted as ${}^P\mathbf{a}'_i$, ${}^P\mathbf{s}'$ and ${}^P\mathbf{I}_G$ in $Px'y'z'$.

B.2 Data for Initial Length Estimation and Input-Shaping Trajectory Planning

Geometrical and inertial properties of the prototype used for Initial Length Estimation (Fig. B.2), defined in Sections 2.1 and 2.3, are summarized in Tables B.3 and B.4, where

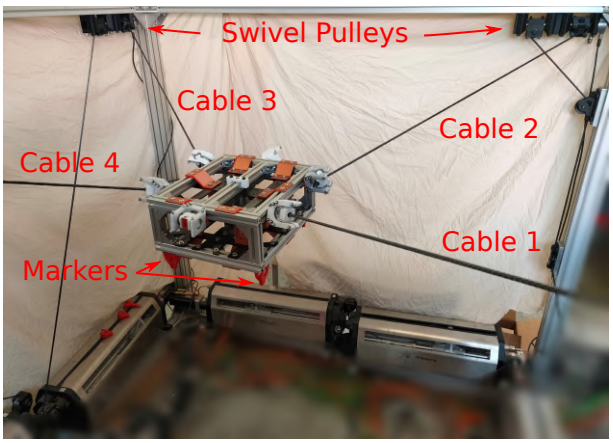


Figure B.1: *UACDPR* Prototype in 2020

Table B.1: Actuators' properties

i	1	2	3	4
$\mathbf{d}_i$ [m]	$\begin{bmatrix} 0.219 \\ -1.316 \\ 0.527 \end{bmatrix}$	$\begin{bmatrix} 2.295 \\ -1.158 \\ 0.521 \end{bmatrix}$	$\begin{bmatrix} 2.153 \\ 0.973 \\ 0.560 \end{bmatrix}$	$\begin{bmatrix} 0.0532 \\ 0.796 \\ 0.532 \end{bmatrix}$
r_i [m]	0.025	0.025	0.025	0.025
${}^P\mathbf{a}'_i$ [m]	$\begin{bmatrix} -0.144 \\ -0.219 \\ 0.264 \end{bmatrix}$	$\begin{bmatrix} 0.115 \\ -0.233 \\ 0.270 \end{bmatrix}$	$\begin{bmatrix} 0.142 \\ 0.220 \\ 0.266 \end{bmatrix}$	$\begin{bmatrix} -0.120 \\ 0.236 \\ 0.266 \end{bmatrix}$
$\mathbf{x}_i$	$\mathbf{j}$	$-\mathbf{i}$	$-\mathbf{j}$	$\mathbf{i}$
$\mathbf{y}_i$	$-\mathbf{k}$	$-\mathbf{k}$	-$\mathbf{k}$	-$\mathbf{k}$
$\mathbf{z}_i$	$-\mathbf{i}$	$-\mathbf{j}$	$\mathbf{i}$	$\mathbf{j}$

Table B.2: Platform inertial properties

m [Kg]	${}^P\mathbf{I}_G$ [Kg$\cdot$m^2]	${}^P\mathbf{s}'$ [m]
8	$\begin{bmatrix} 0.1338 & 0.0059 & 0.0021 \\ 0.0059 & 0.1814 & -0.0055 \\ 0.0021 & -0.0055 & 0.2602 \end{bmatrix}$	$\begin{bmatrix} 0.002 \\ -0.002 \\ 0.200 \end{bmatrix}$

$\mathbf{i} = [1;0;0]^T$, $\mathbf{j} = [0;1;0]^T$, and $\mathbf{k} = [0;0;1]^T$, and the only external load applied to the robot *EE* is gravity, thus $\mathbf{e}' = \mathbf{s}'$, $\boldsymbol{\phi} = -mg\mathbf{k}$ and $\boldsymbol{\mu} = \mathbf{0}_{3\times3}$. The coordinates of $\mathbf{a}'_i$, $\mathbf{s}'$ and $\mathbf{I}_G$ are constant in the *EE* frame, and denoted as ${}^P\mathbf{a}'_i$, ${}^P\mathbf{s}'$ and ${}^P\mathbf{I}_G$ in $Px'y'z'$.

Figure B.2: *UACDPR* Prototype in 2018

Table B.3: Actuators' properties

i	1	2	3
$\mathbf{d}_i$ [m]	$\begin{bmatrix}-2.030\\-0.170\\0.806\end{bmatrix}$	$\begin{bmatrix}0.066\\0.920\\0.757\end{bmatrix}$	$\begin{bmatrix}-2.043\\2.241\\0.738\end{bmatrix}$
r_i [m]	0.025	0.025	0.025
${}^P\mathbf{a}'_i$ [m]	$\begin{bmatrix}0.020\\-0.287\\0.250\end{bmatrix}$	$\begin{bmatrix}0.251\\0.153\\0.250\end{bmatrix}$	$\begin{bmatrix}-0.211\\0.153\\0.250\end{bmatrix}$
$\mathbf{x}_i$	$\mathbf{j}$	$-\mathbf{i}$	$-\mathbf{j}$
$\mathbf{y}_i$	$-\mathbf{k}$	$-\mathbf{k}$	$\mathbf{k}$
$\mathbf{z}_i$	$-\mathbf{i}$	$-\mathbf{j}$	$-\mathbf{i}$

Table B.4: Platform inertial properties

m [Kg]	${}^P\mathbf{I}_G$ [Kg$\cdot$m^2]	${}^P\mathbf{s}'$ [m]
8	$\begin{bmatrix}0.1338 & 0.0059 & 0.0021\\0.0059 & 0.1814 & -0.0055\\0.0021 & -0.0055 & 0.2602\end{bmatrix}$	$\begin{bmatrix}0.000\\0.000\\0.182\end{bmatrix}$

B.3 Data for Rest-to-Rest Trajectory Planning

Geometrical and inertial properties of the prototype used for Initial Length Estimation (Fig. B.2), defined in Sections 2.1 and 2.3, are summarized in Tables B.3 and B.4, where $\mathbf{i} = [1;0;0]^T$, $\mathbf{j} = [0;1;0]^T$, and $\mathbf{k} = [0;0;1]^T$, and the only external load applied to the robot *EE* is gravity, thus $\mathbf{e}' = \mathbf{s}'$, $\boldsymbol{\phi} = -mg\mathbf{k}$ and $\boldsymbol{\mu} = \mathbf{0}_{3\times3}$. The coordinates of $\mathbf{a}'_i$, $\mathbf{s}'$ and $\mathbf{I}_G$ are constant in the *EE* frame, and denoted as ${}^P\mathbf{a}'_i$, ${}^P\mathbf{s}'$ and ${}^P\mathbf{I}_G$ in $Px'y'z'$.

Table B.5: Actuation unit properties

i	1	2	3
$\mathbf{d}_i$ [m]	$\begin{bmatrix} 0.160 \\ -0.835 \\ -0.025 \end{bmatrix}$	$\begin{bmatrix} 2.175 \\ 0.180 \\ -0.035 \end{bmatrix}$	$\begin{bmatrix} 0.260 \\ 1.290 \\ -0.043 \end{bmatrix}$
r_i [m]	0.025	0.025	0.025
${}^P\mathbf{a}'_i$ [m]	$\begin{bmatrix} 0 \\ -0.267 \\ 0.270 \end{bmatrix}$	$\begin{bmatrix} 0.231 \\ 0.133 \\ 0.270] \end{bmatrix}$	$\begin{bmatrix} -0.231 \\ 0.133 \\ 0.270 \end{bmatrix}$
$\mathbf{x}_i$	$\mathbf{j}$	$-\mathbf{i}$	$-\mathbf{j}$
$\mathbf{y}_i$	$-\mathbf{i}$	$-\mathbf{j}$	$\mathbf{i}$
$\mathbf{z}_i$	$\mathbf{k}$	$\mathbf{k}$	$\mathbf{k}$

Table B.6: Platform properties

m [Kg]	${}^P\mathbf{I}_G$ [Kg$\cdot$m^2]	${}^P\mathbf{r}'$ [m]
8	$\begin{bmatrix} 0.14 & 0 & 0 \\ 0 & 0.14 & 0 \\ 0 & 0 & 0.216 \end{bmatrix}$	$\begin{bmatrix} 0 \\ 0 \\ 0.182 \end{bmatrix}$

Appendix C

Additional Experimental Results

This appendix collects additional experimental results which, because of their size, could impair the reading of this manuscript.

C.1 Natural Frequency Compuation

Tables provide the *EE* static configuration $\boldsymbol{\zeta}_0$, the corresponding cable lengths $\mathbf{l}_0$, the array of modeled natural frequencies $\mathbf{f}$, the array of experimental natural frequencies $\mathbf{f}^\star$, the standard deviation obtained in experimental frequency determination $\boldsymbol{\sigma}^\star$, and finally the relative estimation error between experimental and modeled natural frequencies $\Delta\mathbf{f}\%$. Tables C.1 and C.2 are dedicated to a 4-cable UACDPR, Tables C.3 and C.4 to a 3-cable UACDPR and Tables C.5 and C.6 to a 2-cable UACDPR.

Table C.1: Experimental results on 4-cable *UACDPR- Part 1*

Exp.	$\boldsymbol{\zeta}_0$ [m, rad]	$\mathbf{l}_0$ [m]	$\mathbf{f}$ [Hz]	$\mathbf{f}^\star$ [Hz]	$\boldsymbol{\sigma}^\star$ [Hz]	$\Delta\mathbf{f}$%
1	1.28, −0.19, −0.92, −0.00, −0.05, 0.14	1.78, 1.67, 1.74, 1.80	1.06, 1.24	1.06, 1.23	0.01, 0.02	0.65, −0.38
2	1.23, −0.65, −0.97, −0.40, −0.03, 0.10	1.59, 1.57, 2.09, 2.07	0.97, 1.17	0.99, 1.18	0.03, 0.01	2.27, 0.91
3	1.63, −0.62, −0.99, −0.30, −0.23, 0.23	1.87, 1.39, 1.94, 2.27	0.82, 1.19	0.84, 1.25	0.00, 0.01	2.06, 4.68
4	1.72, −0.24, −0.99, 0.01, −0.37, 0.17	2.06, 1.53, 1.67, 2.13	1.10, 1.15	−, 1.16	−, 0.03	−, 1.36
5	1.65, 0.24, −0.97, 0.27, −0.28, 0.04	2.26, 1.82, 1.43, 1.92	0.93, 1.21	0.95, 1.21	0.01, 0.01	1.99, −0.49
6	1.22, 0.27, −0.96, 0.38, −0.00, 0.11	2.06, 2.00, 1.59, 1.62	0.99, 1.19	0.99, 1.18	0.02, 0.01	0.27, −1.26
7	0.78, 0.26, −0.88, 0.38, 0.27, 0.05	1.84, 2.18, 1.78, 1.34	1.08, 1.26	1.09, 1.25	0.03, 0.01	0.35, −0.77
8	0.73, −0.16, −0.88, 0.06, 0.32, 0.06	1.54, 2.00, 1.98, 1.50	1.23, 1.25	1.22, −	0.01, −	−1.07, −
9	0.69, −0.61, −0.87, −0.24, 0.31, 0.02	1.29, 1.87, 2.26, 1.76	0.98, 1.30	1.00, 1.29	0.02, 0.01	1.73, −0.65
10	1.09, −0.16, −0.61, 0.03, 0.06, 0.03	1.51, 1.60, 1.61, 1.49	1.37, 1.62	1.36, 1.61	0.01, 0.00	−0.78, −0.85
11	1.01, −0.59, −0.56, −0.25, 0.06, 0.06	1.20, 1.46, 1.90, 1.68	1.33, 1.64	1.37, 1.64	0.01, 0.01	2.66, −0.56
12	1.51, −0.53, −0.54, −0.19, −0.13, 0.19	1.56, 1.13, 1.63, 1.91	1.31, 1.72	1.30, 1.72	0.00, 0.04	−0.75, 0.05
13	1.84, −0.13, −0.47, 0.04, −0.39, 0.11	1.92, 1.18, 1.20, 1.91	1.46, 2.03	1.46, 2.03	0.02, 0.03	−0.24, −0.28
14	1.79, 0.32, −0.54, 0.29, −0.34, 0.09	2.18, 1.57, 0.99, 1.77	1.24, 1.75	1.24, 1.74	0.00, 0.00	−0.09, −0.82
15	1.30, 0.33, −0.61, 0.37, −0.05, 0.13	1.94, 1.79, 1.27, 1.43	1.37, 1.46	1.41, 1.42	0.04, 0.03	3.03, −2.99
16	0.78, 0.31, −0.67, 0.35, 0.21, 0.12	1.75, 2.09, 1.65, 1.15	1.17, 1.49	1.16, 1.49	0.01, 0.02	−0.98, 0.20
17	0.67, −0.10, −0.69, 0.05, 0.33, 0.12	1.44, 1.95, 1.89, 1.31	1.35, 1.54	1.34, 1.54	0.03, 0.01	−0.64, −0.04
18	0.67, −0.64, −0.72, −0.25, 0.30, 0.03	1.15, 1.78, 2.20, 1.68	1.08, 1.47	1.08, 1.46	0.01, 0.01	−0.23, −0.71

Table C.2: Experimental results on 4-cable *UACDPR- Part 2*

Exp.	$\boldsymbol{\zeta}_0$ [m, rad]	$\mathbf{l}_0$ [m]	$\mathbf{f}$ [Hz]	$\mathbf{f}^\star$ [Hz]	$\boldsymbol{\sigma}^\star$ [Hz]	$\Delta\mathbf{f}$%
19	1.09, −0.15, −0.32, 0.03, 0.05, 0.07	1.36, 1.47, 1.47, 1.33	1.79, 2.31	1.77, 2.29	0.00, 0.04	−1.17, −0.59
20	1.14, −0.59, −0.37, −0.22, −0.01, 0.05	1.17, 1.27, 1.75, 1.65	1.64, 2.02	1.66, 1.98	0.00, 0.06	1.07, −2.25
21	1.57, −0.59, −0.38, −0.23, −0.17, 0.10	1.49, 0.98, 1.57, 1.91	1.61, 1.99	1.61, 1.98	0.01, 0.04	−0.13, −0.47
22	1.70, −0.19, −0.33, 0.00, −0.25, 0.10	1.74, 1.11, 1.20, 1.78	1.68, 2.39	1.68, 2.39	0.00, 0.06	−0.20, −0.36
23	1.66, 0.21, −0.39, 0.16, −0.20, 0.00	1.96, 1.45, 0.98, 1.64	1.40, 2.17	1.39, –	0.00, –	−0.71, –
24	1.26, 0.25, −0.40, 0.26, −0.03, 0.14	1.76, 1.65, 1.20, 1.31	1.71, 1.85	1.69, 1.87	0.01, 0.03	−0.86, 0.91
25	0.79, 0.29, −0.43, 0.27, 0.16, 0.16	1.61, 1.96, 1.54, 0.99	1.43, 1.91	1.43, 1.92	0.00, 0.03	0.34, 0.76
26	0.63, −0.21, −0.45, 0.02, 0.32, 0.07	1.19, 1.82, 1.85, 1.20	1.56, 2.08	1.54, 2.08	0.00, 0.03	−1.49, 0.04
27	0.58, −0.59, −0.46, −0.17, 0.34, 0.05	0.95, 1.74, 2.10, 1.46	1.39, 1.94	1.39, 1.94	0.00, 0.02	0.11, −0.09
28	1.18, −0.20, −0.17, 0.00, 0.01, 0.05	1.33, 1.34, 1.38, 1.36	1.99, 2.95	2.00, 2.91	0.00, 0.08	0.28, −1.34
29	1.19, −0.61, −0.24, −0.20, −0.02, 0.06	1.14, 1.17, 1.68, 1.64	1.90, 2.39	1.94, 2.40	0.01, 0.02	1.99, 0.48
30	1.62, −0.64, −0.22, −0.20, −0.14, 0.15	1.48, 0.82, 1.53, 1.93	1.77, 2.52	1.81, 2.60	0.01, 0.03	2.60, 3.14
31	1.69, −0.19, −0.23, 0.01, −0.21, 0.11	1.70, 1.06, 1.16, 1.74	1.82, 2.72	1.82, 2.73	0.00, 0.07	0.09, 0.28
32	1.63, 0.25, −0.19, 0.16, −0.15, 0.04	1.91, 1.40, 0.85, 1.54	1.73, 2.77	1.73, –	0.00, –	−0.21, –
33	1.26, 0.26, −0.19, 0.20, 0.00, 0.07	1.69, 1.58, 1.08, 1.23	2.00, 2.61	1.99, –	0.03, –	−0.66, –
34	0.74, 0.35, −0.21, 0.26, 0.15, 0.14	1.56, 1.96, 1.47, 0.80	1.76, 2.47	1.85, –	0.01, –	5.15, –
35	0.67, −0.08, −0.19, 0.03, 0.21, 0.13	1.17, 1.76, 1.68, 0.99	1.90, 2.86	1.92, 2.90	0.01, 0.05	1.02, 1.19
36	0.59, −0.57, −0.29, −0.14, 0.29, 0.06	0.84, 1.68, 2.03, 1.37	1.60, 2.38	1.59, 2.36	0.00, 0.04	−0.92, −0.67

Table C.3: Experimental results on 3-cable *UACDPR- Part 1*

Exp.	$\boldsymbol{\zeta}_0$ [m, rad]	$\mathbf{l}_0$ [m]	$\mathbf{f}$ [Hz]	$\mathbf{f}^\star$ [Hz]	$\boldsymbol{\sigma}^\star$ [Hz]	$\Delta\mathbf{f}$%
37	1.61, −0.33, −0.49, −0.20, −0.29, −0.07	1.66, 1.19, 1.41	1.37, 1.66, 2.01	1.35, 1.67, 1.98	0.00, 0.02, 0.06	−1.90, 0.28, −1.10
38	1.47, −0.63, −0.49, −0.36, −0.23, −0.05	1.44, 1.13, 1.69	1.43, 1.49, 1.82	1.45, 1.46, 1.81	0.04, 0.04, 0.02	1.73, −1.90, −0.24
39	1.73, −0.06, −0.53, −0.07, −0.35, −0.08	1.89, 1.32, 1.21	1.27, 1.72, 1.96	1.25, 1.74, –	0.00, 0.01, –	−1.83, 0.78, –
40	1.70, 0.09, −0.31, −0.01, −0.27, −0.09	1.88, 1.32, 0.98	1.24, 2.08, 2.64	1.21, 2.08, 2.61	0.00, 0.06, 0.07	−2.80, 0.12, −1.22
41	1.45, −0.31, −0.33, −0.15, −0.18, −0.08	1.49, 1.20, 1.38	1.33, 1.83, 2.61	1.31, 1.84, 2.60	0.00, 0.02, 0.05	−1.76, 0.24, −0.27
42	1.25, −0.71, −0.39, −0.36, −0.14, −0.05	1.19, 1.20, 1.79	1.37, 1.48, 2.10	1.37, –, 2.11	0.02, –, 0.04	−0.34, –, 0.47
43	1.22, −0.67, −0.20, −0.27, −0.12, −0.06	1.10, 1.15, 1.70	1.47, 1.74, 2.81	1.46, 1.76, 2.78	0.01, 0.02, 0.10	−0.75, 1.04, −1.02
44	1.45, −0.24, −0.16, −0.10, −0.15, −0.09	1.47, 1.15, 1.26	1.39, 2.25, 3.41	1.37, 2.26, –	0.00, 0.06, –	−1.29, 0.22, –
45	1.69, 0.01, −0.10, −0.04, −0.22, −0.08	1.77, 1.19, 0.94	1.43, 2.59, 3.54	1.43, 2.66, –	0.00, 0.06, –	0.23, 3.00, –

Table C.4: Experimental results on 3-cable *UACDPR- Part 2*

Exp.	$\boldsymbol{\zeta}_0$ [m, rad]	$\mathbf{l}_0$ [m]	$\mathbf{f}$ [Hz]	$\mathbf{f}^\star$ [Hz]	$\boldsymbol{\sigma}^\star$ [Hz]	$\Delta\mathbf{f}$%
46	$1.87, -0.56, -0.13, -0.20, -0.32, 0.01$	$1.65, 0.71, 1.37$	$1.77, 1.95,$ 3.09	$1.78, 1.96,$ 3.06	$0.02, 0.03,$ 0.07	$0.74, 0.67,$ -1.19
47	$1.89, -0.64, -0.25, -0.29, -0.38, -0.00$	$1.67, 0.73, 1.48$	$1.61, 1.67,$ 2.50	$1.63, -,$ 2.48	$0.03, -,$ 0.07	$1.37, -,$ -0.72
48	$1.89, -0.64, -0.46, -0.37, -0.44, -0.02$	$1.74, 0.89, 1.57$	$1.35, 1.57,$ 1.90	$1.36, 1.55,$ 1.89	$0.01, 0.02,$ 0.06	$0.29, -0.93,$ -0.45

Table C.5: Experimental results on 2-cable *UACDPR- Part 1*

Exp.	$\boldsymbol{\zeta}_0$ [m, rad]	$\mathbf{l}_0$ [m]	$\mathbf{f}$ [Hz]	$\mathbf{f}^\star$ [Hz]	$\boldsymbol{\sigma}^\star$ [Hz]	$\Delta\mathbf{f}$%
49	$1.12, -0.21, -0.97, 0.02, 0.07, -0.14$	$1.72, 1.84$	$0.44, 0.89,$ $1.41, 1.57$	$0.44, 0.89,$ $1.41, 1.57$	$0.00, 0.03,$ $0.04, 0.02$	$1.14, -0.84,$ $0.03, -0.50$
50	$0.84, -0.53, -0.97, -0.14, 0.21, -0.13$	$1.45, 2.17$	$0.43, 0.87,$ $1.33, 1.55$	$0.44, 0.86,$ $1.35, 1.54$	$0.00, 0.01,$ $0.05, 0.02$	$1.23, -1.31,$ $1.52, -0.61$

Table C.6: Experimental results on 2-cable *UACDPR- Part 2*

Exp.	$\boldsymbol{\zeta}_0$ [m, rad]	$\mathbf{l}_0$ [m]	$\mathbf{f}$ [Hz]	$\mathbf{f}^\star$ [Hz]	$\boldsymbol{\sigma}^\star$ [Hz]	$\Delta\mathbf{f}$%
51	1.53, 0.27, −0.91, 0.26, −0.13, −0.12	2.17, 1.40	0.44, 0.86, 1.37, 1.57	0.44, 0.85, 1.34, 1.57	0.00, 0.01, 0.05, 0.02	−0.41, −1.09, −2.30, −0.16
52	1.18, −0.13, −0.70, 0.06, 0.04, −0.15	1.61, 1.59	0.49, 0.91, 1.76, 1.89	0.48, 0.89, 1.77, 1.90	0.01, 0.00, 0.04, 0.05	−1.73, −1.97, 0.31, 0.46
53	0.88, −0.49, −0.75, −0.09, 0.17, −0.14	1.31, 2.00	0.47, 0.89, 1.65, 1.71	0.48, 0.90, −, 1.70	0.00, 0.02, −, 0.03	0.93, 1.13, −, −0.73
54	1.58, 0.32, −0.77, 0.27, −0.15, −0.12	2.15, 1.24	0.47, 0.86, 1.53, 1.66	0.47, 0.87, 1.52, 1.65	0.00, 0.03, 0.01, 0.00	0.50, 0.43, −0.90, −0.34
55	1.20, −0.11, −0.49, 0.06, 0.03, −0.15	1.52, 1.45	0.54, 0.92, 1.99, 2.41	0.55, 0.91, 2.00, −	0.01, 0.00, 0.05−	1.77, −1.43, 0.15, −
56	0.85, −0.52, −0.48, −0.08, 0.16, −0.14	1.09, 1.90	0.54, 0.91, 1.95, 2.28	0.55, 0.91, 1.96, 2.26	0.00, 0.03, 0.03, 0.07	1.38, −0.03, 0.33, −0.79
57	1.51, 0.24, −0.44, 0.18, −0.07, −0.13	1.91, 1.06	0.55, 0.91, 1.97, 2.35	0.55, 0.90, 1.98, 2.31	0.00, 0.00, 0.03, 0.06	0.05, −1.46, 0.33, −1.31
58	1.22, −0.10, −0.18, 0.06, 0.03, −0.15	1.41, 1.29	0.65, 1.00, 2.60, 3.61	0.66, 0.99, −, −	0.00, 0.00, −, −	1.67, −0.70, −, −
59	0.83, −0.55, −0.21, −0.05, 0.13, −0.15	0.91, 1.84	0.64, 0.98, 2.41, 3.25	0.64, 0.98, −, −	0.00, 0.01, −, −	0.76, −0.17, −, −
60	1.60, 0.33, −0.17, 0.17, −0.06, −0.14	1.94, 0.80	0.65, 0.98, 2.41, 3.26	0.66, 0.96, 2.43, −	0.00, 0.01, 0.06, −	2.20, −2.46, 0.77, −

List of Author Publications

1. E. Idá, A. Berti, T. Bruckmann, and M. Carricato, "Rest-to-Rest trajectory planning for planar underactuated cable-driven parallel robots," in *Cable-Driven Parallel Robots* (C. Gosselin, P. Cardou, T. Bruckmann, and A. Pott, eds.), (Cham), pp. 207–218, Springer, 2018.

2. E. Idá, J.-P. Merlet, and M. Carricato, "Automatic self-calibration of suspended under-actuated cable-driven parallel robot using incremental measurements," in *Cable-Driven Parallel Robots* (A. Pott and T. Bruckmann, eds.), (Cham), pp. 333–344, Springer International Publishing, 2019.

3. E. Idà, T. Bruckmann, and M. Carricato, "Rest-to-rest trajectory planning for underactuated cable-driven parallel robots," *IEEE Transactions on Robotics*, vol. 35, pp. 1338–1351, Dec 2019.

4. E. Idà, S. Briot, and M. Carricato, "Robust trajectory planning of under-actuated cable-driven parallel robot with 3 cables," in *Advances in Robot Kinematics 2020* (J. Lenarčič and B. Siciliano, eds.), (Cham), pp. 65–72, Springer International Publishing, 2021.

5. E. Idà, D. Marian, and M. Carricato, "A deployable cable-driven parallel robot with large rotational capabilities for laser-scanning applications," *IEEE Robotics and Automation Letters*, vol. 5, no. 3, pp. 4140–4147, 2020.

6. F. Zaccaria, S. Briot, M. T. Chikhaoui, E. Idà, and M. Carricato, "An analytical formulation for the geometrico-static problem of continuum planar parallel robots," in *ROMANSY 23 - Robot Design, Dynamics and Control* (G. Venture, J. Solis, Y. Takeda, and A. Konno, eds.), (Cham), pp. 512–520, Springer International Publishing, 2021.

7. E. Idà, S. Briot, and M. Carricato, "Natural Oscillations of Underactuated Cable-Driven Parallel Robots." Submitted to *IEEE Transaction on Robotics*.

8. S. Olivieri, F. Martini, E. Idà, and M. Carricato, "Towards Multidisciplinary Engineering Curriculum Design: a Pilot Study to Teach Control Education in Mechanical Engineering with MATLAB/SIMULINK and Arduino." Proceedings of the 48th SEFI Conference, 2020, IN PRESS.

List of Figures

Bibliography

[1] C. M. Gosselin and J. . Hamel, "The agile eye: a high-performance three-degree-of-freedom camera-orienting device," in *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, pp. 781–786 vol.1, 1994.

[2] K. Miura, H. Furuya, and K. Suzuki, "Variable geometry truss and its application to deployable truss and space crane arm," *Acta Astronautica*, vol. 12, no. 7, pp. 599 – 607, 1985.

[3] S. E. Lansdberger and T. B. Sheridan, "A new design for parallel link manipulators," in *Proc. Sys. Man. and Cybernetics Conf., Tucson, AZ*, pp. 812–814, 1985.

[4] Y. Su and B. Duan, "The mechanical design and kinematics accuracy analysis of a fine tuning stable platform for the large spherical radio telescope," *Mechatronics*, vol. 10, no. 7, pp. 819–834, 2000.

[5] S. Kawamura, W. Choe, S. Tanaka, and H. Kino, "Development of an ultrahigh speed robot falcon using parallel wire drive systems," *Journal of the Robotics Society of Japan*, vol. 15, no. 1, pp. 82–89, 1997.

[6] L. Gagliardini, S. Caro, M. Gouttefarde, and A. Girin, "Discrete reconfiguration planning for cable-driven parallel robots," *Mechanism and Machine Theory*, vol. 100, pp. 313 – 337, 2016.

[7] P. Miermeister, M. Lächele, R. Boss, C. Masone, C. Schenk, J. Tesch, M. Kerger, H. Teufel, A. Pott, and H. H. Bülthoff, "The cablerobot simulator large scale motion platform based on cable robot technology," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 3024–3029, 2016.

[8] A. Ming and T. Higuchi, "Study on multiple degree-of-freedom positioning mechanism using wires. i: Concept, design and control," *International Journal of the Japan Society for Precision Engineering*, vol. 28, no. 2, pp. 131–138, 1994.

[9] A. Pott, H. Mütherich, W. Kraus, V. Schmidt, P. Miermeister, and A. Verl, "Ipanema: A family of cable-driven parallel robots for industrial applications," in *Cable-Driven Parallel Robots* (T. Bruckmann and A. Pott, eds.), pp. 119–134, Springer, 2013.

[10] T. Bruckmann, M. Hiller, and D. Schramm, "An active suspension system for simulation of ship maneuvers in wind tunnels," in *New Trends in Mechanism Science* (D. Pisla, M. Ceccarelli, M. Husty, and B. Corves, eds.), (Dordrecht), pp. 537–544, Springer Netherlands, 2010.

[11] T. Bruckmann, C. Sturm, L. Fehlberg, and C. Reichert, "An energy-efficient wire-based storage and retrieval system," in *2013 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, pp. 631–636, July 2013.

[12] A. Pott, *Cable-driven parallel robots: theory and application*, vol. 120. Springer, 2018.

[13] M. Conconi and M. Carricato, "A new assessment of singularities of parallel kinematic chains," *IEEE Transactions on Robotics*, vol. 25, pp. 757–770, Aug 2009.

[14] M. Carricato and J.-P. Merlet, "Geometrico-static analysis of under-constrained cable-driven parallel robots," in *Advances in Robot Kinematics: Motion in Man and Machine*, pp. 309–319, Springer, 2010.

[15] M. Carricato and J. Merlet, "Stability Analysis of Underconstrained Cable-Driven Parallel Robots," *IEEE Transactions on Robotics*, vol. 29, pp. 288–296, Feb 2013.

[16] J.-P. Merlet, "On the redundancy of cable-driven parallel robots," in *New Trends in Mechanism and Machine Science* (P. Flores and F. Viadero, eds.), (Cham), pp. 31–39, Springer International Publishing, 2015.

[17] X. Jiang and C. Gosselin, "Dynamic point-to-point trajectory planning of a three-dof cable-suspended parallel robot," *IEEE Transactions on Robotics*, vol. 32, no. 6, pp. 1550–1557, 2016.

[18] X. Jiang, E. Barnett, and C. Gosselin, "Dynamic point-to-point trajectory planning beyond the static workspace for six-dof cable-suspended parallel robots," *IEEE Transactions on Robotics*, vol. 34, no. 3, pp. 781–793, 2018.

[19] X. Jiang, E. Barnett, and C. Gosselin, "Periodic trajectory planning beyond the static workspace for 6-dof cable-suspended parallel robots," *IEEE Transactions on Robotics*, vol. 34, no. 4, pp. 1128–1140, 2018.

[20] G. Mottola, C. Gosselin, and M. Carricato, "Dynamically feasible periodic trajectories for generic spatial three-degree-of-freedom cable-suspended parallel robots," *Journal of Mechanisms and Robotics*, vol. 10, no. 3, 2018.

[21] N. Zhang, W. Shang, and S. Cong, "Dynamic trajectory planning for a spatial 3-dof cable-suspended parallel robot," *Mechanism and Machine Theory*, vol. 122, pp. 177 – 196, 2018.

[22] G. Mottola, C. Gosselin, and M. Carricato, "Dynamically feasible motions of a class of purely-translational cable-suspended parallel robots," *Mechanism and Machine Theory*, vol. 132, pp. 193–206, 2019.

[23] T. Heyden and C. Woernle, "Dynamics and flatness-based control of a kinematically undetermined cable suspension manipulator," *Multibody System Dynamics*, vol. 16, p. 155, Aug 2006.

[24] M. Yamamoto, N. Yanai, and A. Mohri, "Trajectory control of incompletely restrained parallel-wire-suspended mechanism based on inverse dynamics," *IEEE Transactions on Robotics*, vol. 20, pp. 840–850, Oct 2004.

[25] N. Michael, S. Kim, J. Fink, and V. Kumar, "Kinematics and statics of cooperative multi-robot aerial manipulation with cables," in *ASME 2009 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, pp. 83–91, American Society of Mechanical Engineers, 2009.

[26] D. Cunningham and H. H. Asada, "The winch-bot: A cable-suspended, underactuated robot utilizing parametric self-excitation," in *2009 IEEE International Conference on Robotics and Automation*, pp. 1844–1850, May 2009.

[27] N. Zoso and C. Gosselin, "Point-to-point motion planning of a parallel 3-dof underactuated cable-suspended robot," in *2012 IEEE International Conference on Robotics and Automation*, pp. 2325–2330, May 2012.

[28] J. Park, O. Kwon, and J. H. Park, "Anti-sway trajectory generation of incompletely restrained wire-suspended system," *Journal of Mechanical Science and Technology*, vol. 27, pp. 3171–3176, Oct 2013.

[29] S. W. Hwang, J.-H. Bak, J. Yoon, and J. H. Park, "Oscillation reduction and frequency analysis of under-constrained cable-driven parallel robot with three cables," *Robotica*, p. 1–21, 2019.

[30] S. W. Hwang, J.-H. Bak, J. Yoon, J. H. Park, and J.-O. Park, "Trajectory generation to suppress oscillations in under-constrained cable-driven parallel robots," *Journal of Mechanical Science and Technology*, vol. 30, pp. 5689–5697, Dec 2016.

[31] G. Abbasnejad and M. Carricato, "Real solutions of the direct geometrico-static problem of under-constrained cable-driven parallel robots with 3 cables: a numerical investigation," *Meccanica*, vol. 47, no. 7, pp. 1761–1773, 2012.

[32] M. Carricato, "Direct geometrico-static problem of underconstrained cable-driven parallel robots with three cables," *Journal of Mechanisms and Robotics*, vol. 5, no. 3, p. 031008, 2013.

[33] G. Abbasnejad and M. Carricato, "Direct geometrico-static problem of underconstrained cable-driven parallel robots with n cables," *IEEE Transactions on Robotics*, vol. 31, pp. 468–478, April 2015.

[34] A. Berti, J.-P. Merlet, and M. Carricato, "Solving the direct geometrico-static problem of underconstrained cable-driven parallel robots by interval analysis," *The International Journal of Robotics Research*, vol. 35, no. 6, pp. 723–739, 2016.

[35] M. Carricato, G. Abbasnejad, and D. Walter, "Inverse Geometrico-Static Analysis of Under-Constrained Cable-Driven Parallel Robots with Four Cables," in *Latest Advances in Robot Kinematics* (J. Lenarcic and M. Husty, eds.), (Dordrecht), pp. 365–372, Springer Netherlands, 2012.

[36] M. Carricato, "Inverse geometrico-static problem of underconstrained cable-driven parallel robots with three cables," *Journal of Mechanisms and Robotics*, vol. 5, no. 3, p. 031002, 2013.

[37] R. de Rijk, M. Rushton, and A. Khajepour, "Out-of-plane vibration control of a planar cable-driven parallel robot," *IEEE/ASME Transactions on Mechatronics*, vol. 23, no. 4, pp. 1684–1692, 2018.

[38] M. Zarei, A. Aflakian, A. Kalhor, and M. T. Masouleh, "Oscillation damping of nonlinear control systems based on the phase trajectory length concept: An experimental case study on a cable-driven parallel robot," *Mechanism and Machine Theory*, vol. 126, pp. 377 – 396, 2018.

[39] M. R. Jafari Harandi, H. Damirchi, S. a. Khalilpour seyedi, and H. D. Taghirad, "Point-to-point motion control of an underactuated planar cable driven robot," in *2019 27th Iranian Conference on Electrical Engineering (ICEE)*, pp. 979–984, 2019.

[40] S. Lefrançois and C. Gosselin, "Point-to-point motion control of a pendulum-like 3-dof underactuated cable-driven robot," in *2010 IEEE International Conference on Robotics and Automation*, pp. 5187–5193, 2010.

[41] L. Barbazza, D. Zanotto, G. Rosati, and S. K. Agrawal, "Design and optimal control of an underactuated cable-driven micro-macro robot," *IEEE Robotics and Automation Letters*, vol. 2, pp. 896–903, April 2017.

[42] L. Scalera, A. Gasparetto, and D. Zanotto, "Design and experimental validation of a 3-dof underactuated pendulum-like robot," *IEEE/ASME Transactions on Mechatronics*, vol. 25, no. 1, pp. 217–228, 2020.

[43] G. Oriolo and Y. Nakamura, "Control of mechanical systems with second-order nonholonomic constraints: underactuated manipulators," in *30th IEEE Conference on Decision and Control*, pp. 2398–2403, Dec 1991.

[44] E. Idá, J.-P. Merlet, and M. Carricato, "Automatic self-calibration of suspended under-actuated cable-driven parallel robot using incremental measurements," in *Cable-Driven Parallel Robots* (A. Pott and T. Bruckmann, eds.), (Cham), pp. 333–344, Springer International Publishing, 2019.

[45] E. Idà, T. Bruckmann, and M. Carricato, "Rest-to-rest trajectory planning for underactuated cable-driven parallel robots," *IEEE Transactions on Robotics*, vol. 35, pp. 1338–1351, Dec 2019.

[46] E. Idà, S. Briot, and M. Carricato, "Robust trajectory planning of under-actuated cable-driven parallel robot with 3 cables," in *Advances in Robot Kinematics 2020* (J. Lenarčič and B. Siciliano, eds.), (Cham), pp. 65–72, Springer International Publishing, 2021.

[47] I. A. Bonev and J. Ryu, "A new approach to orientation workspace analysis of 6-dof parallel manipulators," *Mechanism and machine theory*, vol. 36, no. 1, pp. 15–28, 2001.

[48] A. Pott, "Influence of pulley kinematics on cable-driven parallel robots," in *Latest Advances in Robot Kinematics* (J. Lenarcic and M. Husty, eds.), pp. 197–204, Dordrecht: Springer, 2012.

[49] G. H. Golub and C. F. Van Loan, "Matrix computations, 4th," *Johns Hopkins*, 2013.

[50] A. Berti, J.-P. Merlet, and M. Carricato, "Solving the direct geometrico-static problem of 3-3 cable-driven parallel robots by interval analysis: Preliminary results," in *Cable-Driven Parallel Robots*, pp. 251–268, Springer, 2013.

[51] D. Surdilovic, J. Radojicic, and J. Krüger, "Geometric stiffness analysis of wire robots: A mechanical approach," in *Cable-Driven Parallel Robots* (T. Bruckmann and A. Pott, eds.), (Berlin, Heidelberg), pp. 389–404, Springer Berlin Heidelberg, 2013.

[52] Z. Cui, X. Tang, S. Hou, and H. Sun, "Research on controllable stiffness of redundant cable-driven parallel robots," *IEEE/ASME Transactions on Mechatronics*, vol. 23, pp. 2390–2401, Oct 2018.

[53] S. Behzadipour and A. Khajepour, "Stiffness of Cable-based Parallel Manipulators With Application to Stability Analysis," *Journal of Mechanical Design*, vol. 128, pp. 303–310, 04 2005.

[54] A. H. Nayfeh and D. T. Mook, *Nonlinear oscillations*. John Wiley & Sons, 2008.

[55] J.-P. Merlet, *Parallel robots*, vol. 74. Springer, 2012.

[56] J.-P. Merlet, "Analysis of the influence of wires interference on the workspace of wire robots," in *On Advances in Robot Kinematics* (J. Lenarčič and C. Galletti, eds.), (Dordrecht), pp. 211–218, Springer Netherlands, 2004.

[57] S. Perreault, P. Cardou, C. M. Gosselin, and M. J.-D. Otis, "Geometric determination of the interference-free constant-orientation workspace of parallel cable-driven mechanisms," *Journal of Mechanisms and Robotics*, vol. 2, no. 3, p. 031016, 2010.

[58] A. Martin, S. Caro, and P. Cardou, "Geometric determination of the cable-cylinder interference regions in the workspace of a cable-driven parallel robot," in *Cable-Driven Parallel Robots*, pp. 117–127, Springer, 2018.

[59] Y. Wischnitzer, N. Shvalb, and M. Shoham, "Wire-driven parallel robot: Permitting collisions between wires," *The International Journal of Robotics Research*, vol. 27, no. 9, pp. 1007–1026, 2008.

[60] C. B. Pham, S. H. Yeo, G. Yang, and I.-M. Chen, "Workspace analysis of fully restrained cable-driven manipulators," *Robotics and Autonomous Systems*, vol. 57, no. 9, pp. 901–912, 2009.

[61] M. Gouttefarde and C. M. Gosselin, "Analysis of the wrench-closure workspace of planar parallel cable-driven mechanisms," *IEEE Transactions on Robotics*, vol. 22, no. 3, pp. 434–445, 2006. cited By 200.

[62] M. Gouttefarde, D. Daney, and J. Merlet, "Interval-analysis-based determination of the wrench-feasible workspace of parallel cable-driven robots," *IEEE Transactions on Robotics*, vol. 27, pp. 1–13, Feb 2011.

[63] J. Pusey, A. Fattah, S. Agrawal, and E. Messina, "Design and workspace analysis of a 6-6 cable-suspended parallel robot," *Mechanism and Machine Theory*, vol. 39, no. 7, pp. 761 – 778, 2004.

[64] E. Stump and V. Kumar, "Workspaces of Cable-Actuated Parallel Manipulators," *Journal of Mechanical Design*, vol. 128, pp. 159–167, 07 2005.

[65] G. Barrette and C. M. Gosselin, "Determination of the dynamic workspace of cable-driven planar parallel mechanisms," *Journal of Mechanical Design*, vol. 127, no. 2, pp. 242–248, 2005.

[66] A. A. Kumar, J.-F. Antoine, P. Zattarin, and G. Abba, "Workspace analysis of a 4 cable-driven spatial parallel robot," in *ROMANSY 22 – Robot Design, Dynamics and Control* (V. Arakelian and P. Wenger, eds.), (Cham), pp. 204–212, Springer International Publishing, 2019.

[67] P. Cardou, S. Bouchard, and C. Gosselin, "Kinematic-sensitivity indices for dimensionally nonhomogeneous jacobian matrices," *IEEE Transactions on Robotics*, vol. 26, no. 1, pp. 166–173, 2010.

[68] G. Mottola, C. Gosselin, and M. Carricato, "Effect of actuation errors on a purely-translational spatial cable-driven parallel robot," in *2019 IEEE 9th Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER)*, pp. 701–707, 2019.

[69] J. P. Merlet, "Jacobian, Manipulability, Condition Number, and Accuracy of Parallel Robots," *Journal of Mechanical Design*, vol. 128, pp. 199–206, 06 2005.

[70] S. Briot, A. Pashkevich, and D. Chablat, "On the Optimal Design of Parallel Robots Taking Into Account Their Deformations and Natural Frequencies," in *33rd Mechanisms and Robotics Conference, Parts A and B*, vol. 7 of *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, pp. 367–376, 08 2009.

[71] K. Kozak, I. Ebert-Uphoff, and W. Singhose, "Locally Linearized Dynamic Analysis of Parallel Manipulators and Application of Input Shaping to Reduce Vibrations ," *Journal of Mechanical Design*, vol. 126, pp. 156–168, 03 2004.

[72] S. Briot and M. Gautier, "Global identification of joint drive gains and dynamic parameters of parallel robots," *Multibody System Dynamics*, vol. 33, no. 1, pp. 3–26, 2015.

[73] W. Khalil and E. Dombre, *Modeling, identification and control of robots*. Butterworth-Heinemann, 2004.

[74] R. Chellal, E. Laroche, L. Cuvillon, and J. Gangloff, "An identification methodology for 6-dof cable-driven parallel robots parameters application to the inca 6d robot," in *Cable-Driven Parallel Robots* (T. Bruckmann and A. Pott, eds.), (Berlin, Heidelberg), pp. 301–317, Springer Berlin Heidelberg, 2013.

[75] R. Chellal, L. Cuvillon, and E. Laroche, "Model identification and vision-based h∞ position control of 6-dof cable-driven parallel robots," *International Journal of Control*, vol. 90, no. 4, pp. 684–701, 2017.

[76] P. Tempel, P. Herve, O. Tempier, M. Gouttefarde, and A. Pott, "Estimating inertial parameters of suspended cable-driven parallel robots — use case on cogiro," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 6093–6098, May 2017.

[77] E. Picard, S. Caro, F. Claveau, and F. Plestan, "Pulleys and force sensors influence on payload estimation of cable-driven parallel robots," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1429–1436, Oct 2018.

[78] W. Kraus, V. Schmidt, P. Rajendra, and A. Pott, "System identification and cable force control for a cable-driven parallel robot with industrial servo drives," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 5921–5926, May 2014.

[79] M. Gobbi, G. Mastinu, and G. Previati, "A method for measuring the inertia properties of rigid bodies," *Mechanical Systems and Signal Processing*, vol. 25, no. 1, pp. 305 – 318, 2011.

[80] L. Tang and W.-B. Shangguan, "An improved pendulum method for the determination of the center of gravity and inertia tensor for irregular-shaped bodies," *Measurement*, vol. 44, no. 10, pp. 1849 – 1859, 2011.

[81] Cong Bang Pham, Guilin Yang, and Song Huat Yeo, "Dynamic analysis of cable-driven parallel mechanisms," in *Proceedings, 2005 IEEE/ASME International Conference on Advanced Intelligent Mechatronics.*, pp. 612–617, July 2005.

[82] S. Van Huffel and J. Vandewalle, *The total least squares problem: computational aspects and analysis*. SIAM, 1991.

[83] M. Gautier and W. Khalil, "Exciting trajectories for the identification of base inertial parameters of robots," *International Journal of Robotics Research*, vol. 11, no. 4, pp. 362–375, 1992.

[84] J. Jin and N. Gans, "Parameter identification for industrial robots with a fast and robust trajectory design approach," *Robotics and Computer-Integrated Manufacturing*, vol. 31, pp. 21 – 29, 2015.

[85] V. Bonnet, P. Fraisse, A. Crosnier, M. Gautier, A. González, and G. Venture, "Optimal exciting dance for identifying inertial parameters of an anthropomorphic structure," *IEEE Transactions on Robotics*, vol. 32, no. 4, pp. 823–836, 2016.

[86] J. Swevers, W. Verdonck, and J. De Schutter, "Dynamic model identification for industrial robots," *IEEE Control Systems Magazine*, vol. 27, no. 5, pp. 58–71, 2007.

[87] M. Gautier, P. O. Vandanjon, and C. Presse, "Identification of inertial and drive gain parameters of robots," in *Proceedings of 1994 33rd IEEE Conference on Decision and Control*, vol. 4, pp. 3764–3769, 1994.

[88] C. Presse and M. Gautier, "New criteria of exciting trajectories for robot identification," in *Proceedings IEEE International Conference on Robotics and Automation*, vol. 3, pp. 907–912, 1993.

[89] T. Katsumata, B. Navarro, V. Bonnet, P. Fraisse, A. Crosnier, and G. Venture, "Optimal exciting motion for fast robot identification. application to contact painting tasks with estimated external forces," *Robotics and Autonomous Systems*, vol. 113, pp. 149 – 159, 2019.

[90] J. Nocedal and S. Wright, *Numerical optimization*. Springer Science & Business Media, 2006.

[91] A. Nubiola, M. Slamani, A. Joubair, and I. A. Bonev, "Comparison of two calibration methods for a small industrial robot based on an optical cmm and a laser tracker," *Robotica*, vol. 32, no. 3, p. 447–466, 2014.

[92] D. Daney, N. Andreff, G. Chabert, and Y. Papegay, "Interval method for calibration of parallel robots: Vision-based experiments," *Mechanism and Machine Theory*, vol. 41, no. 8, pp. 929 – 944, 2006.

[93] J.-P. Merlet, "Direct kinematics of cdpr with extra cable orientation sensors: The 2 and 3 cables case with perfect measurement and ideal or elastic cables," in *Cable-Driven Parallel Robots* (C. Gosselin, P. Cardou, T. Bruckmann, and A. Pott, eds.), (Cham), pp. 180–191, Springer, 2018.

[94] A. Pott, "An algorithm for real-time forward kinematics of cable-driven parallel robots," in *Advances in Robot Kinematics: Motion in Man and Machine* (J. Lenarcic and M. M. Stanisic, eds.), pp. 529–538, Dordrecht: Springer, 2010.

[95] P. H. Borgstrom, B. L. Jordan, B. J. Borgstrom, M. J. Stealey, G. S. Sukhatme, M. A. Batalin, and W. J. Kaiser, "Nims-pl: A cable-driven robot with self-calibration capabilities," *IEEE Transactions on Robotics*, vol. 25, no. 5, pp. 1005–1015, 2009.

[96] D. Lau, "Initial length and pose calibration for cable-driven parallel robots with relative length feedback," in *Cable-Driven Parallel Robots* (C. Gosselin, P. Cardou, T. Bruckmann, and A. Pott, eds.), (Cham), pp. 140–151, Springer International Publishing, 2018.

[97] P. Miermeister, A. Pott, and A. Verl, "Auto-calibration method for overconstrained cable-driven parallel robots," in *ROBOTIK 2012; 7th German Conference on Robotics*, pp. 1–6, VDE, 2012.

[98] A. Pott, H. Mütherich, W. Kraus, V. Schmidt, P. Miermeister, and A. Verl, "Ipanema: A family of cable-driven parallel robots for industrial applications," in *Cable-Driven Parallel Robots* (T. Bruckmann and A. Pott, eds.), (Berlin, Heidelberg), pp. 119–134, Springer Berlin Heidelberg, 2013.

[99] J. A. d. Sandretto, G. Trombettoni, D. Daney, and G. Chabert, "Certified calibration of a cable-driven robot using interval contractor programming," in *Computational Kinematics* (F. Thomas and A. Perez Gracia, eds.), (Dordrecht), pp. 209–217, Springer Netherlands, 2014.

[100] L. Biagiotti and C. Melchiorri, *Trajectory planning for automatic machines and robots*. Springer Science & Business Media, 2008.

[101] R. Seifried, *Dynamics of underactuated multibody systems: modeling, control and optimal design*. Springer, 2013.

[102] A. Isidori, *Nonlinear control systems*. Springer, 2013.

[103] N. C. Singer and W. P. Seering, "Preshaping Command Inputs to Reduce System Vibration," *Journal of Dynamic Systems, Measurement, and Control*, vol. 112, pp. 76–82, 03 1990.

[104] W. Singhose, E. Crain, and W. Seering, "Convolved and simultaneous two-mode input shapers," *IEE Proceedings - Control Theory and Applications*, vol. 144, pp. 515–520, Nov 1997.

[105] W. E. Singhose, L. J. Porter, T. D. Tuttle, and N. C. Singer, "Vibration Reduction Using Multi-Hump Input Shapers," *Journal of Dynamic Systems, Measurement, and Control*, vol. 119, pp. 320–326, 06 1997.

[106] J. M. Hollerbach, "Dynamic Scaling of Manipulator Trajectories," *Journal of Dynamic Systems, Measurement, and Control*, vol. 106, pp. 102–106, 03 1984.

[107] K. Graichen, V. Hagenmeyer, and M. Zeitz, "A new approach to inversion-based feedforward control design for nonlinear systems," *Automatica*, vol. 41, no. 12, pp. 2033 – 2041, 2005.

[108] K. Graichen and M. Zeitz, "Feedforward control design for finite-time transition problems of non-linear mimo systems under input constraints," *International Journal of Control*, vol. 81, no. 3, pp. 417–427, 2008.

[109] J. S. Short, A. N. Poo, C. Y. Lai, P. Y. Tao, and M. H. Ang, "A generalized underactuated robot system inversion method using hamiltonian formalism," in *2015 IEEE International Conference on Advanced Intelligent Mechatronics (AIM)*, pp. 1415–1420, July 2015.

[110] A. Piazzi and A. Visioli, "Optimal noncausal set-point regulation of scalar systems," *Automatica*, vol. 37, no. 1, pp. 121 – 127, 2001.

[111] J. Bobrow, S. Dubowsky, and J. Gibson, "Time-optimal control of robotic manipulators along specified paths," *The International Journal of Robotics Research*, vol. 4, no. 3, pp. 3–17, 1985.

[112] L. F. Shampine, J. Kierzenka, and M. W. Reichelt, "Solving boundary value problems for ordinary differential equations in Matlab with bvp4c," *Tutorial notes*, vol. 2000, pp. 1–27, 2000.

[113] U. M. Ascher and L. R. Petzold, *Computer methods for ordinary differential equations and differential-algebraic equations*. Siam, 1998.