Research paper

Static workspace computation for underactuated cable-driven parallel robots☆

Edoardo Ida' *, Marco Carricato

Department of Industrial Engineering, University of Bologna, 40137 Bologna, Italy

ARTICLE INFO

ABSTRACT

Cable-Driven Parallel Robots (*CDPRs*) move an end-effector (*EE*) using cables arranged in a parallel fashion. If a *CDPR* employs fewer cables than its *EE* degrees of freedom (*DoFs*), the robot is generally underactuated and underconstrained. Consequently, only a subset of the *EE DoFs* can be assigned for trajectory planning purposes, and the *EE* pose cannot be inferred by only relying on forward kinematics. Consequently, it is not trivial to assess the robot workspace (*WS*), even though *WS* computation is of paramount importance in analyzing the robot's performance. This paper introduces a novel algorithm for the computation of the reachable static *WS* of generic underactuated *CDPRs*, namely the set of *EE* positions that are statically attainable with at least one orientation and characterized by positive and bounded cable tensions. The algorithm leverages a novel geometrico-static problem, which, given a candidate *EE* position, seeks an orientation satisfying a stable static equilibrium characterized by a desired tension distribution.

1. Introduction

A cable-driven parallel robot (*CDPR*) is a parallel manipulator whose end-effector (*EE*) is moved by cables instead of rigid-link legs. *CDPRs* present specific advantages over traditional parallel robots, such as a remarkably large reachable workspace [1], achieved by spooling long cables on servo-actuated winches. 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 system control.

A *CDPR* is generally underactuated (*UA*), or kinematically deficient, if it is equipped with a number n of actuated cables that is smaller than the *EE* degrees of freedom (*DoFs*), n_d. Each actuated (taut) cable imposes a geometrical constraint on the pose of the *EE*, ζ, and said constraint depends on the length of the cable. Since the number of geometrical constraints equals the number n of actuated cables, and $n < n_d$, the forward geometric problem of the *UACDPR* is intrinsically underdetermined [2]. Namely, an infinity of *EE* poses corresponds to a set of assigned cable lengths. Actuation deficiency also complicates trajectory planning [3]: since the rank of the robot geometric Jacobian is at most $n < n_d$, not every *EE DoF* is instantaneously controllable [4]. In a parallel robot, actuation deficiency strictly implies constraint deficiency [5]: the *EE* is underconstrained and thus preserves mobility even if its actuators are locked. The *EE* mechanical equilibrium must be considered to compute physically attainable *EE* configurations. Despite these limitations, *UACDPRs* are suitable for applications where enhanced workspace accessibility and mechanical simplicity are favored, and limited mobility and control of their *EE* is acceptable [6–8]. On the other hand, research on both theoretical and practical aspects of *UACDPRs* is at an early stage. Geometrico-static problems [2,9], and the stability of their solutions [4,10,11] were

---

investigated, as well as trajectory planning to limit the *EE* natural oscillations while moving between equilibrium set-points [3,12]. Feedback strategies compensating for forward kinematics deficiencies were developed [13,14], and feedback control was proposed for regulating the *EE* task while damping and removing unwanted oscillations [15–18].

A robotic manipulator's workspace (*WS* in short) can be defined as *the set of configurations that can be reached by its EE*, given some physical limits of the specific system at hand. It is a crucial property that defines the robot's possible applications, and its analysis needs to be carried out at the design stage of any manipulator to determine the *EE* motion range and possibly its performance in a determined set of configurations. It is often of practical interest to limit *WS* evaluation or analysis to a subset of all the attainable *EE* configurations if the robot is built for a specific purpose or has a non-conventional architecture [19]. For example, the *constant-orientation* or *translational WS* is the set of all possible locations of the *EE* reference point that can be reached with a given orientation [20], the *orientation WS* is the set of all possible orientations of the *EE* that can be reached with a given position of the reference point [21], and the *reachable WS* is the set of all possible locations of the *EE* reference point that can be reached with at least one orientation [22]. To assess if a configuration of the *EE* belongs to the *WS*, several criteria may be considered, such as cables not interfering with themselves and the environment [23–25]. Since cables can only pull, a criterion specific to *CDPRs* is wrench closure or feasibility [26–28], namely the ability of the *EE* to counteract a prescribed set of wrenches with cables exerting a bounded positive tension. If the set of wrenches is limited to the gravitational one, the constraint becomes *static-feasibility* [29], and allows one to determine the workspace where the manipulator can be brought to rest. On the other hand, recent studies also considered inertial actions among the wrenches applied to the *EE*, thus defining the concept of *dynamic WS* [30,31].

To the knowledge of the authors, only two works attempted to propose a methodology for the computation of the workspace of *UACDPRs*, focusing on the computation of the reachable and statically-feasible (often referred to as *static*) workspace (*RSW*) of a 4-cable system [32] and of a 3-cable system [22]. In [32], the authors discretized the translational task space and looked for any *EE* orientation satisfying static equilibrium for a prescribed position by solving an inverse geometrico-static problem (*GSP*); if an orientation was found, and the cable tensions were within bounds, the *EE* position was included in the *RSW*. In the authors' opinion, this algorithm is limited by two main factors: (i) not all the geometrico-statically feasible poses of the *EE* of a *UACDPR* are stable [2,10] and thus reachable; (ii) no particular orientation is found by the algorithm, as long as it is feasible: this hinders the practical use of said algorithm, where some theoretically feasible orientations are practically non-attainable [33], due to lack of continuity between adjacent poses or cable becoming slack due to feedback errors. The authors of [22], instead, focused on computing the boundaries of 3-cable-robot *RSW*; to this end, they proposed a boundary formulation and a model for translating such a formulation in linear equalities and inequalities, which are then numerically solved; the models developed by the authors were tailored to 3-cable systems, and may not be directly applicable to other architectures.

This paper proposes a novel algorithm for the computation of the *RSW* of *any UACDPR*. To this end, the translational task space is discretized to devise a set of *candidate* positions, which are added to the *RSW* iff (i) an *EE* orientation, compatible with static equilibrium, is computed, (ii) the resulting *EE* equilibrium configuration is stable, and (iii) the cable tensions, computed in said equilibrium, are within given positive bounds. To this end, a modified inverse *GSP* is formulated for each candidate position as a set of non-linear equations. The classical inverse *GSP*, requiring as many assigned task-space variables as cables, is modified to require only a given position, as well as some parameters influencing cable tensions: *a single orientation with a desired cable tension distribution* can thus be found for a given assigned position, which in turns allows one to compute a series of statically reachable set-points which can be used both for performance quantification or trajectory planning for *any UACDPR*. Workspace computation and performance quantification are often used to drive optimal design, and are performed iteratively; trajectory planning, on the other hand, may need to run in real-time. Thus, three techniques are employed to speed up computation. On the one hand, wrench feasibility and equilibrium stability are checked only if a statically feasible orientation exists, thus are performed as "ex-post" checks, instead of being considered as constraints in the solution process. Then, candidate *RSW* positions are scanned according to a predefined rationale, so that each tentative solution is the actual solution to the nearest solved problem: this strategy ensures the highest chance to find continuous solutions throughout the workspace. Finally, the problem Jacobian is derived, and embedded into the solution procedure. Our method has three distinctive features with respect to the state of the art: (i) it can deal with any *UACDPR* architecture, (ii) an orientation is computed to achieve a specific tension distribution (if possible), and (iii) *WS* poses are continuous.

This article is an extension of the conference paper presented in [34], where only the 4-cable robot was investigated. With respect to [34], the current paper presents the following original contributions:

- the workspace-computation algorithm can handle UACDPRs with any number of cables comprised between 2 and 5;
- an application to trajectory planning is proposed, to show the algorithm potentiality;
- an analytical formulation of the Jacobian of the modified inverse GSP is presented in the Appendix, which can drastically increase the computational efficiency of the workspace-computation algorithm.

In the following, Section 2 introduces the inverse *GSP* of *UACDPRs*, and its original modification. Section 3 details the workspace computation algorithm, and Section 4 proposes its application to 2- to 5-cable *UACDPR*. Section 5 provides an example of our algorithm merits for trajectory planning, and Section 6 draws conclusions.

2. Inverse geometrico-static problem

This Section briefly recalls the classical geometrico-static model of an *UACDPR* (borrowed from [10]), and introduces the modified inverse *GSP*, to be used for workspace computation.
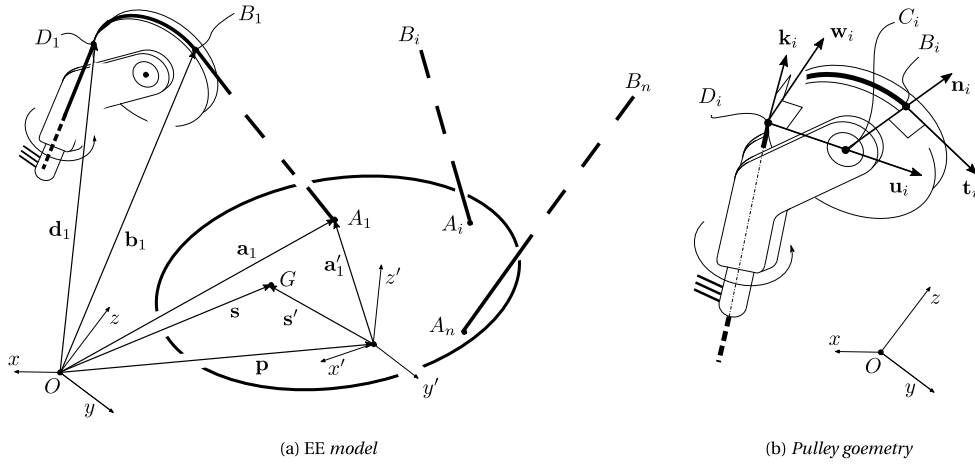
(a) EE *model*   (b) *Pulley goemetry*

Fig. 1. *UACDPR geometric model.*

A mobile *EE* is linked to the frame with $n < \mu$ extendable cables, coiled and uncoiled by motorized winches ($\mu = 6$ in $SE(3)$, $\mu = 3$ in $SE(2)$); the difference between the number μ of *DoFs* and the number n of cables is denoted by $\kappa = \mu - n$. $Oxyz$ is a fixed reference system attached to the robot frame, and $Px'y'z'$ is a moving reference system attached to the *EE*; the pose of the latter is described by the position vector p of P and the rotation matrix $\mathbf{R}$, which in turn is parametrized by a minimal set of angles ϵ, namely $\mathbf{R} = \mathbf{R}(\epsilon)$. The *EE* generalized coordinates are thus $\zeta = \begin{bmatrix} p^T & \epsilon^T \end{bmatrix}^T$.

Cables are modeled as massless straight segments.[1] Each cable is guided into the workspace by a swiveling pulley, which can rotate with respect to a support hinged to the robot frame (Fig. 1). Point D_i, B_i, and A_i are, respectively, the i-th cable entry point into the pulley, its exit point from the pulley, and its attachment point on the *EE*. The position vector from O to D_i is constant and denoted by vector d_i in $Oxyz$ (Fig. 1(a)). If the position vector from P to A_i in $Px'y'z'$ is $^P a'_i$, the position vector a_i from O to A_i in $Oxyz$ is given by:

$$a_i = p + a'_i = p + \mathbf{R}\,^P a'_i, \qquad a'_i \overset{\Delta}{=} \mathbf{R}\,^P a'_i \tag{1}$$

The position vector of B_i from O, denoted by b_i, depends on the platform pose and the pulley model [4,12]. The constant unit vector k_i denotes the swivel pulley axis in $Oxyz$, whereas the unit vector u_i is directed from D_i to C_i, the pulley center, and $w_i = k_i \times u_i$, with $\times$ denoting the vector product. Unit vector n_i points from C_i to B_i, and unit vector $t_i = w_i \times n_i$ is directed as the cable exiting the pulley (Fig. 1(b)). Assuming the pulley is frictionless, the cable lies on the plane passing through unit vector k_i and point A_i. Accordingly, if r_i is the radius of the i-th pulley, then:

$$b_i = d_i + r_i \left(u_i + n_i \right) \tag{2}$$

If $\rho_i \overset{\Delta}{=} a_i - b_i$, l_i is the cable length comprising the rectilinear part $\|\rho_i\|$ and the arc $\widehat{B_i D_i}$ wrapped on the pulley, and $(\cdot)$ indicates the scalar product, the geometric constraint imposed by the i-th cable on the platform is given by:

$$\rho_i \cdot \rho_i - \left[l_i - \widehat{B_i D_i} \right]^2 = 0 \tag{3}$$

2.1. Inverse geometrico-static problem

The static model is derived considering the external actions acting on the *EE*, namely the cable tensions and the gravitational wrench (Fig. 2). Mechanical equilibrium of the *EE* yields:

$$\Xi^T \tau - f = \mathbf{0}_{6\times 1}, \qquad f = m \begin{bmatrix} g \\ \tilde{s}' g \end{bmatrix} \tag{4}$$

where g is the gravitational acceleration, m and G are the *EE* mass and center of mass, $^P s'$ and $s' = \mathbf{R}\,^P s'$ are the position vectors from P to G respectively in $Px'y'z'$ and $Oxyz$, $(\tilde{\cdot})$ denotes the skew-symmetric representation of a vector, $\mathbf{0}_{6\times 1}$ is a null vector of

---

[1] The cable model is kept as simple as possible, mainly because introducing elasticity or sagging would affect neither the formulation of the modified inverse geometrico-static problem, which will be introduced in Section 2.2, nor the workspace computation algorithm of Section 3, as the static equilibrium equation would remain the same. What would change, indeed, is the analytical formulation of the Jacobian matrix, which may become appreciably more complex (and out of the scope of this paper).
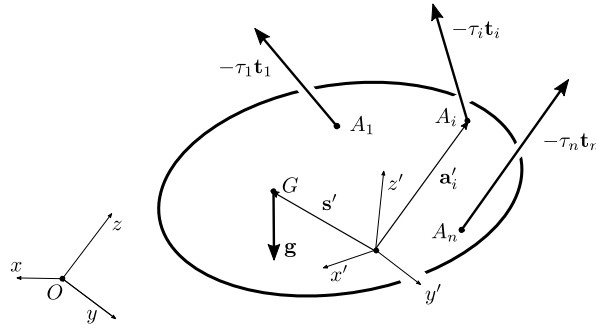
Fig. 2. *EE* static model.

dimension 6, $\boldsymbol{\tau} = [\tau_1, \ldots, \tau_n]^T$ is the array containing the cable tensions, and $\boldsymbol{\Xi}^T \in \mathbb{R}^{6 \times n}$ (usually referred to as structure matrix) is the transpose of the inverse kinematics Jacobian matrix $\boldsymbol{\Xi}$, whose i-th row is [35]:

$$\boldsymbol{\Xi}_i = \begin{bmatrix} \boldsymbol{t}_i^T & -\boldsymbol{t}_i^T \tilde{\boldsymbol{a}}_i' \end{bmatrix} \tag{5}$$

The geometrico-static model is established considering Eqs. (3) for $i = 1, \ldots, n$ and (4):

$$
\begin{cases}
\boldsymbol{\rho}_1 \cdot \boldsymbol{\rho}_1 - \left[l_1 - \widehat{B_1 D_1} \right]^2 &= 0 \\
\qquad\qquad\vdots \\
\boldsymbol{\rho}_n \cdot \boldsymbol{\rho}_n - \left[l_n - \widehat{B_n D_n} \right]^2 &= 0 \\
\boldsymbol{\Xi}^T \boldsymbol{\tau} - \boldsymbol{f} &= \boldsymbol{0}_{6 \times 1}
\end{cases}
\tag{6}
$$

The non-linear system (6) has $n + 6$ equations in $2n + 6$ variables, namely the position and orientation parameters of the *EE*, the n cable lengths, and the n cable tensions. We establish the forward *GSP* by assigning cable lengths and solving for cable tensions and the *EE* pose; alternatively, the inverse *GSP* is defined by assigning a subset of n coordinates of the *EE* pose and solving for cable lengths, cable tensions, and the remaining *EE* coordinates. Both problems are well-posed since they comprise $n + 6$ equations in $n + 6$ unknowns. They may admit multiple real solutions: a solution is acceptable if the equilibrium is stable [10], and cable tensions are strictly positive and bounded within τ_m and τ_M, which are, respectively, the minimum and the maximum tension limit, to be set in advance depending on the application.

Regarding the workspace computation, one can resort to joint-space discretization, compute several possible cable length sets, and use a forward GSP problem to find reachable configurations. Unfortunately, it was shown that, this way, it is statistically unlikely to obtain feasible equilibrium configurations with all cables taut for 4- and 5-cable robots [2]. Alternatively, one can use task-space discretization and solve, at each iteration, a corresponding inverse GSP. However, the classical formulation of the inverse *GSP* has some disadvantages, especially when used for 4- and 5-cable systems. Even though the position of the *EE* is a natural choice as part of the assigned subset of *EE* coordinates, selecting the additional coordinates is cumbersome [36]. In addition, [33] showed that orientation parameters might vary in a minimal range, so it is impractical to *strictly* assign them: the overall equilibrium configuration may not be robust against model and control uncertainties. The authors of [32] proposed only to assign the *EE* position, and to solve the resulting underdetermined system with a non-linear least square optimization, accepting any orientation and cable tensions that would satisfy minimum and maximum tension limits. This strategy, though simple, cannot guarantee to produce *EE* equilibrium poses that are practically usable as set-points for trajectory planning [37] since the equilibrium poses may be unstable, and thus not reachable, and orientation continuity between adjacent poses is not guaranteed. Moreover, this strategy makes the performance quantification of the reachable *EE* position possibly unfair [33], since the random orientation computed by the method influences the performance of the overall configuration. Differently from 4- and 5-cable robots, for 3-cable robots the problem is simplified, since one may only assign the *EE* position [22].

2.2. Modified inverse geometrico-static problem: Spatial case

The limitations outlined above motivated the authors to propose a modified formulation of the inverse *GSP*, where only the *EE* position is assigned, and the *EE* orientation is found so that, at the equilibrium, a desired tension distribution is achieved, when possible. As an additional benefit of this method, the computed equilibrium configurations can be used as practical control set points of the robot, since adjacent poses are continuous and a specified tension distribution is guaranteed.

The static equilibrium in Eq. (4) can be divided into force and moment equilibria:

$$\boldsymbol{\Xi}_F^T \boldsymbol{\tau} - m\boldsymbol{g} = \boldsymbol{0}_{3 \times 1} \tag{7}$$

$$\boldsymbol{\Xi}_M^T \boldsymbol{\tau} - m\tilde{\boldsymbol{s}}' \boldsymbol{g} = \boldsymbol{0}_{3 \times 1} \tag{8}$$

where $\Xi_F^T \in \mathbb{R}^{3 \times n}$ and $\Xi_M^T \in \mathbb{R}^{3 \times n}$ are obtained from the first and the last three rows of Ξ^T, respectively. Then, we seek a tension distribution τ in the form:

$$\tau = \tau_0 + \tau^\perp, \qquad \tau_0 \overset{\Delta}{=} m\Xi_F^{-T} g, \qquad \tau^\perp \overset{\Delta}{=} \Xi_F^{T\perp} \lambda \tag{9}$$

where $\lambda \in \mathbb{R}^{(3-\kappa) \times 1}$ is an array of parameters to be determined, $\Xi_F^{-T} \in \mathbb{R}^{n \times 3}$ is any right inverse of Ξ_F^T and $\Xi_F^{T\perp} \in \mathbb{R}^{n \times (3-\kappa)}$ is any right nullspace of Ξ_F^T, if any exists (see the Introduction of [38] for additional details).[2] A possible formulation of Ξ_F^{-T} and $\Xi_F^{T\perp}$ is provided in Eq. (A.7) of Appendix. The principal advantage of the proposed formulations is their explicit analytical expressions, which can be symbolically differentiated (which is useful for the numerical implementation). Regardless of the number of cables employed, $\Xi_F^T \Xi_F^{-T}$ is the identity matrix of order 3, while a right nullspace resulting in $\Xi_F^T \Xi_F^{T\perp} = \mathbf{0}_{3 \times (3-\kappa)}$ only exists for 4- and 5-cable robots. For the 3-cable case, $n = \kappa = 3$ (and thus $\kappa - 3 = 0$), Ξ_F^T is a square generally-full-rank matrix, Ξ_F^{-T} is a standard inverse matrix, and $\Xi_F^{T\perp}$ is not defined. The tension distribution in Eq. (9) always satisfies force equilibrium, but it may not satisfy moment equilibrium. If we substitute Eq. (9) in Eq. (8), the modified GSP can be formulated as:

$$m \left(\Xi_M^T \Xi_F^{-T} - \tilde{s}' \right) g + \Xi_M^T \Xi_F^{T\perp} \lambda = \mathbf{0}_{3 \times 1} \tag{10}$$

which is a system of 3 non-linear equations in $9 - \kappa$ variables (the EE position and orientation, respectively p and ϵ, and λ), and can be described by:

$$\phi(p, \epsilon, \lambda) = \mathbf{0}_{3 \times 1}, \qquad \phi \overset{\Delta}{=} m \left(\Xi_M^T \Xi_F^{-T} - \tilde{s}' \right) g + \Xi_M^T \Xi_F^{T\perp} \lambda \tag{11}$$

We call *inverse* the problem where the position is assigned, and the orientation and λ are computed as solutions of Eq. (11). When $n = 3$, the problem is square and allows one to compute a discrete set of possible EE orientations [39]. In this case, no tension distribution can be specified, and Eq. (11) can be simplified as:

$$\phi_3(\epsilon) = \mathbf{0}_{3 \times 1}, \qquad \phi_3 \overset{\Delta}{=} m \left(\Xi_M^T \Xi_F^{-T} - \tilde{s}' \right) g \tag{12}$$

When $n = 4$ or 5, the problem is underdetermined and generally admits infinite solutions. The problem becomes square if a pre-defined tension distribution is specified. In practice, one may formulate λ as an explicit function of the EE position and orientation according to several state-of-the-art criteria (see [40], chapter 3, Section 7), depending on the desired tension-distribution performance, namely $\lambda = \lambda(\epsilon)$. This aspect is detailed in Section 4 but, in general, Eq. (11) can be formulated as:

$$\phi_{4,5}(p, \epsilon) = \mathbf{0}_{3 \times 1}, \qquad \phi_{4,5} \overset{\Delta}{=} m \left(\Xi_M^T \Xi_F^{-T} - \tilde{s}' \right) g + \Xi_M^T \Xi_F^{T\perp} \lambda(\epsilon) \tag{13}$$

Eqs. (12) and (13) are two systems of 3 non-linear equations in 3 unknowns, namely the elements of ϵ. Depending on the number of cables n and the formulation used to compute λ, a finite set of solutions may be found numerically, e.g. by a Newton–Raphson algorithm. After a solution is found, its wrench feasibility and equilibrium stability must be assessed. Wrench feasibility requires $\tau_m \preceq \tau \preceq \tau_M$, where the symbol $\preceq$ denotes element-wise inequality between a scalar and a vector quantity. In contrast, equilibrium stability requires the positive sign of the eigenvalues of the so-called *free motion stiffness matrix* $\mathbf{K}_f^\perp$ described in [4]:

$$\mathbf{K}_f^\perp \overset{\Delta}{=} \Xi^{\perp T} (\mathbf{K} + \mathbf{Q}) \Xi^\perp \tag{14}$$

where $\Xi^{\perp T} \in \mathbb{R}^{\kappa \times 6}$ is any left nullspace of Ξ^T ($\Xi^{\perp T} \Xi^T = \mathbf{0}_{\kappa \times n}$) and $\mathbf{K}$ is computed as:

$$\mathbf{K} \overset{\Delta}{=} \sum_{i=1}^n \tau_i \begin{bmatrix} \mathbf{T}_i & -\mathbf{T}_i \tilde{a}_i' \\ \tilde{a}_i' \mathbf{T}_i & -\tilde{a}_i' \mathbf{T}_i \tilde{a}_i' \end{bmatrix} + \sum_{i=1}^n \tau_i \begin{bmatrix} \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \tilde{t}_i \tilde{a}_i' \end{bmatrix} \tag{15}$$

$$\mathbf{T}_i \overset{\Delta}{=} \frac{\|u_i \times n_i\| w_i w_i^T}{u_i \cdot (a_i - d_i)} + \frac{n_i n_i^T}{\|\rho_i\|}, \qquad \mathbf{Q} \overset{\Delta}{=} \begin{bmatrix} \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & -m\tilde{g}\tilde{s}' \end{bmatrix} \tag{16}$$

If the equilibrium configuration of the EE is stable and wrench feasible, cable lengths can be computed through Eq. (3), and the configuration is considered *reachable*. It is worth noting that the solution set may change by changing the formulation of λ, namely the desired tension-distribution performance of the equilibrium configuration.

2.3. Modified inverse geometrico-static problem: Planar case

This section briefly summarizes the difference in the Modified Inverse GSP when the robot is a 2-cable planar UACDPR ($n = 2$) and $\mu = 3$, since vectors and matrices have different sizes compared to the spatial case, and their use and definitions may not be straightforward. All geometric vectors are now 2-dimensional: if motion occurs on the xy plane, with y being the direction of gravity, the third component, in the z direction, is omitted, and $\mathbf{R}$ is a 2×2 rotation matrix about the z axis (see [14] for details on implementation). Consequently, Eqs. (7) and (8) can be rewritten as:

$$\Xi_{F,P}^T \tau - mg = \mathbf{0}_{2 \times 1} \tag{17}$$

---

[2] Please note that the order of operation in transposing a matrix and obtaining its nullspace is not commutative. A matrix $\mathbf{A} \in \mathbb{R}^{b \times c}$, with $b < c$ and $\text{rank}(A) = b$, has a *right* nullspace whose dimension is $c \times (c - b)$, whereas $\mathbf{A}^T$ has a *left* nullspace whose dimension is $(c - b) \times c$.

$$\Xi_{M,P}^T \boldsymbol{\tau} - m\tilde{s}' \boldsymbol{g} = 0 \tag{18}$$

where $\Xi_{F,P}^T \in \mathbb{R}^{2\times 2}$ and $\Xi_{M,P}^T \in \mathbb{R}^{1\times 2}$, since $\Xi^T \in \mathbb{R}^{2\times 3}$. Equivalently to the spatial 3-cable case, $\Xi_{F,P}^T$ is a square generally-full-rank matrix, $\Xi_{F,P}^{-T}$ is a standard inverse matrix, and $\Xi_{F,P}^{T\perp} = \boldsymbol{0}_{2\times 1}$, and thus the modified inverse *GSP* in Eq. (10) becomes:

$$m\left(\Xi_{M,P}^T \Xi_{F,P}^{-T} - \tilde{s}'\right)\boldsymbol{g} = 0 \tag{19}$$

which allows one to compute a discrete set of orientations without the possibility of specifying a tension distribution. The free-motion stiffness matrix $\mathbf{K}_f^\perp$ is now a scalar:

$$K_f^\perp \stackrel{\Delta}{=} \Xi^{\perp T}(\mathbf{K} + \mathbf{Q})\Xi^\perp \tag{20}$$

where $\Xi^{\perp T} \in \mathbb{R}^{1\times 3}$ is any left nullspace of Ξ^T ($\Xi^{\perp T}\Xi^T = \boldsymbol{0}_{1\times 2}$) and $\mathbf{K}, \mathbf{Q} \in \mathbb{R}^{3\times 3}$. Stability is assessed by checking the positive sign of $K_f^\perp$.

3. Workspace computation algorithm

This section details the main aspects of the workspace computation algorithm for *UACDPRs*, namely its aim, advantages, shortcomings, and numerical implementation details. The algorithm is based on discretizing the *EE* positions so that a regular Cartesian grid of nodes covers the potential workspace. Each node is then tested to see whether it belongs to the workspace by first numerically solving Eq. (11) with an iterative scheme, such as the Newton–Raphson one, and then checking if the solution is both wrench-feasible and stable. The main advantage of this strategy is the simplicity of implementation, whereas 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;
- correct boundary representation may involve a large number of nodes.

However, our aim is not the accurate determination of the workspace boundaries or volume, but the quick determination of a finite set of configurations that: (i) are locally continuous with respect to neighboring configurations, so that the *EE* can reach them in practice while statically maintaining cable tensions within given bounds and with a given tension distribution, and (ii) may be assessed for local or global performance evaluation. The main steps for workspace calculation are summarized as follows:

1. lower and upper limits of the *EE* position coordinates, respectively $[x_l, x_u]$, $[y_l, y_u]$, and $[z_l, z_u]$, are assigned in order to define a limiting box where the workspace is investigated;
2. the intervals defined by lower and upper limits are divided by a regular grid of n_g nodes, where n_g is odd (thus, there is a central node in each interval);
3. for each node, a numerical solution of the modified inverse *GSP* in Eq. (11) is sought;
4. if there is a solution, wrench feasibility, and stability are checked;
5. if the checks in item 4 are satisfied, the configuration is added to the workspace;
6. consecutive nodes are checked along predefined directions; if a solution in item 3 is not found or the checks in item 4 are not satisfied, a point outside the workspace is found, workspace exploration is stopped along the current direction, and another direction is explored.

A possible pseudocode of the workspace computation algorithm is given in Alg. 1, where only the exploration of one quadrant is given for brevity's sake. Exploring the other quadrants only requires changing the limits of the for-loops, and does not add relevant information for implementation.

The proposed algorithm relies on the numerical solution of the modified inverse *GSP* in Eq. (11), which is a set of 3 non-linear equations in 3 unknowns. Regardless of the numerical method employed for solving Eq. (11) (e.g. Newton–Raphson, Levenberg–Marquardt, etc.), 3 issues are important for achieving a fast, accurate, and continuous solution to the problem:

- the tension distribution computation method must be continuous for small pose displacements;
- the method needs to be input a tentative solution near the expected one, namely an *initial guess*;
- an analytical formulation of the Jacobian of Eq. (11), namely $\boldsymbol{J} = \partial\boldsymbol{\phi}/\partial\boldsymbol{\epsilon}$, is preferably to be provided.

The first and second issues are related to solution accuracy and continuity. There is a vast literature related to tension distribution algorithms, and their continuity issues (see [40], Chapter 3, Section 7), but several methods, typically characterized by an explicit analytical expression, are continuous (e.g. the barycentric method used in [41]). Then, if $\lambda(\boldsymbol{\epsilon})$ is a continuous and differentiable function, the numerical solution of Eq. (11) is continuous and differentiable as well, if and only if (i) Eq. (11) is not ill-posed, and (ii) a suitable initial guess is provided to the solver; inputting a user-provided tentative solution in the first node, and using the solution to the closest problem in the Euclidean sense for all other nodes[3] should ensure solution continuity in practice: the

---

[3] There are *workspace exploration strategies* that can streamline such a tentative solution selection [42], but they are not investigated here, since they are not fundamental for understanding and implementing the results proposed in this paper.

Algorithm 1: Workspace Computation Algorithm Pseudocode

Data: $[x_l, x_u]$, $[y_l, y_u]$, $[z_l, z_u]$, n_g, τ_m, τ_M, *UACDPR* geometry and inertial parameters
Result: Set of configurations ζ belonging to the workspace

if n_g *is even* **then**
  | $n_g = n_g - 1$
end
$x_c = (x_l + x_u)/2$ `/* x-coordinate of the central node */`
$y_c = (y_l + y_u)/2$ `/* y-coordinate of the central node */`
$z_c = (z_l + z_u)/2$ `/* z-coordinate of the central node */`
$\epsilon_g = \mathbf{0}_3 \times 1$ `/* orientation initial guess for the first node */`
$k = 1$
Initialize an empty workspace   `/* Computation of the workspace in the quadrant */`
                                                          `/* defined by the intervals `$[x_c, x_u]$`, `$[y_c, y_u]$`, and `$[z_c, z_u]$` */`

for x **from** x_c **to** x_u **by** $(x_u - x_l)/n_g$ **do**
  for y **from** y_c **to** y_u **by** $(y_u - y_l)/n_g$ **do**
    for z **from** z_c **to** z_u **by** $(z_u - z_l)/n_g$ **do**
      $p_k = [x, y, z]^T$
      if *Workspace is not empty* **then**
        | $\epsilon_g \leftarrow \epsilon_i$ so that $\|p_k - p_i\|$ is minimum, and p_i is the *i*-th point of the workspace, with $i < k$ `/* Update initial guess */`
      end
      $\epsilon_k \leftarrow$ Solve Eq. (11) for a given p_k, with initial guess ϵ_g
      if ϵ_k *is a solution of Eq.* (11) **and** $\zeta_k = \left[p_k^T, \epsilon_k^T\right]^T$ *is wrench feasible and stable* **then**
        Assign ζ_k to the workspace
        $k = k + 1$
      else
        Exit the innermost for-loop   `/* The workspace border is met, and the exploration in a given direction is`
         `terminated */`
      end
    end
  end
end

(a) *Feasible orientation, the* EE *is oriented as desired*

(b) *Unfeasible orientation, the* EE *is tilted with respect to the desired working mode, and two cables (dashed) interfere with the platform*
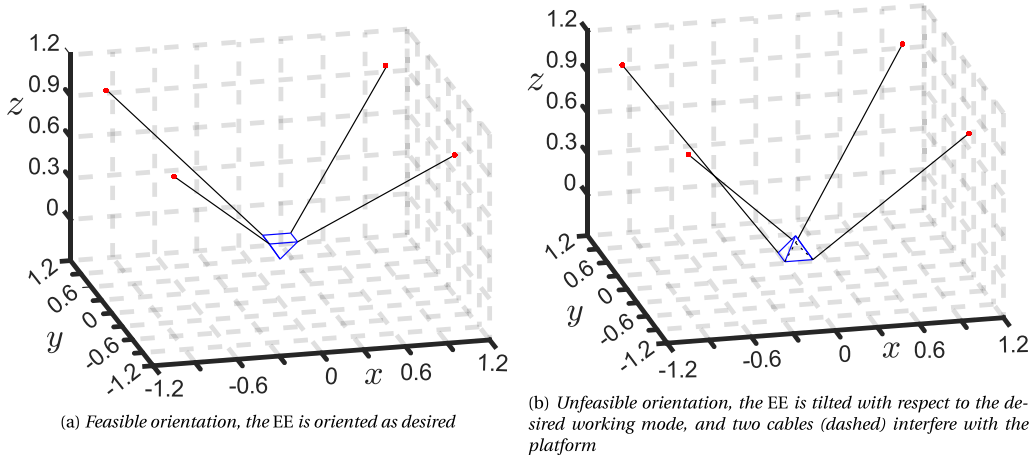
Fig. 3. Stable and wrench-feasible equilibrium for $\mathbf{p} = [0, 0, 0]^T$ (units are in meters). The *EE* is schematized in blue, cables are in black, and cable exit points from the frame are in red. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

configurations computed in adjacent nodes are expected to be continuous, since each modified inverse *GSP* uses a neighboring problem solution as the tentative solution for the numerical solver. However, no analytical proof of continuity is available, and we cannot guarantee that discontinuities do not occur in very special situations. Section 5 presents a numerical example showing the continuity of the results of our algorithm in a typical practical case. It should be noted that this algorithm requires that the solution to the very first modified inverse *GSP* is selected carefully: depending on this choice, different solutions can be propagated throughout the rest of the algorithm, corresponding to non-practically usable *EE* working modes (see Fig. 3). Many solutions to the first problem, among which the user can select the desired one, can be obtained with algorithms for randomly selecting various initial guesses (e.g. MATLAB *multistart*).

Providing an analytical formulation of the Jacobian of Eq. (11), though limitedly affects the accuracy or continuity of the problem solution, is important for computing performance since finite-difference approximations significantly increase the computation time (see [43], Chapter 8, Section 1). The details about the computation of $\mathbf{J}$ are provided in Appendix.
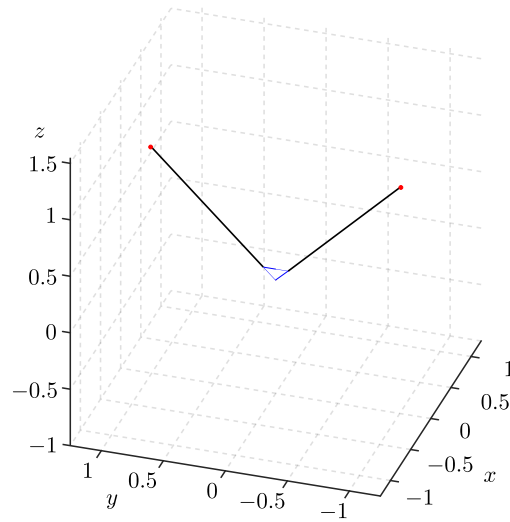
Fig. 4. Schematics of the 2-cable *UACDPR* for $\mathbf{p} = [0,0,0]^T$ (units are in meters). The *EE* is schematized in blue, cables are in black, and swiveling pulleys are in red. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 1
2-cable *UACDPR* geometric parameters.

i	1	2
$\mathbf{d}_i$ [m]	$\begin{bmatrix} 0 \\ -1 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}$
r_i [m]	0.0	0.0
$^P\mathbf{a}'_i$ [m]	$\begin{bmatrix} 0 \\ -0.1 \\ 0.1 \end{bmatrix}$	$\begin{bmatrix} 0 \\ 0.1 \\ 0.1 \end{bmatrix}$

4. Workspace computation examples

In this section, the method for workspace computation outlined in Section 3 is applied to a 2-cable robot, a 3-cable robot, two 4-cable robots, and a 5-cable robot. For illustrative purposes, suitable tension-distribution methods are chosen for the 4- and 5-cable robots. The *EE* mass is $10\,\mathrm{kg}$, and its center-of-mass-position in $Px'y'z'$ is $^P\mathbf{s}' = [0,0,0.05]^T$ m. The lower and upper limits of the *EE* position coordinates, needed as input parameters by the workspace algorithm, are taken as $x_l = \min\left(d_{i,x}\right)$, $x_u = \max\left(d_{i,x}\right)$, $y_l = \min\left(d_{i,y}\right)$, $y_u = \max\left(d_{i,y}\right)$, $z_l = -1\,\mathrm{m}$, $z_u = \max\left(d_{i,z} - 0.16\right)$ m, for $i = 1,\dots,n$, which result in a prism including each cable entry point in its pulley, an arbitrary inferior limit for the z coordinate, and a safety superior limit for the z coordinates so that cables are never all coplanar. The geometric and inertial parameters of the robots used in the following subsections are inspired by several configurations of the IRMAL@B prototype of the University of Bologna [44] n_g is set to 101 for enhanced border-reconstruction accuracy, and, for each architecture, the workspace for two sets of cable-tension limits is evaluated to show how they impact the reachable-workspace size. The limits are $\tau'_m = 30\,\mathrm{N}$ and $\tau'_M = 100\,\mathrm{N}$, or $\tau''_m = 5\,\mathrm{N}$ and $\tau''_M = 500\,\mathrm{N}$, where the latter define a broader tension range. The direction of each swiveling-pulley axis $\mathbf{k}_i$ ($i = 1,\dots,n$) is $[0,0,1]^T$ for all robots. The workspace computation algorithm, with a Newton–Raphson method for the solution of the modified inverse *GSP* (see [43], Chapter 11, Section 1), was implemented in Matlab on a PC with a 10th generation i7 Intel processor and 16 Gb of RAM.

4.1. 2-cable UACDPR

The geometric parameters and a schematic of the 2-cable robot can be found in Table 1 and Fig. 4. The results of the workspace computation are shown in Fig. 5, where the two tension limit sets $[\tau'_m, \tau'_M]$ and $[\tau''_m, \tau''_M]$ were considered. Fig. 5(b) shows a larger workspace than the one in Fig. 5(a) since the most limiting factors defining workspace boundaries appear to be the tension limits, but, when these limits are large enough, the suspended 2-cable *UACDPR* can reach most of the area below its pulleys. The computational time was 5.8 s for the workspace in Fig. 7(b), and 11.4 s for the workspace in Fig. 7(a).

4.2. 3-cable UACDPR

The geometric parameters and a schematic of the 3-cable robot can be found in Table 2 and Fig. 6. The results of the workspace computation are shown in Fig. 7, where the two tension limit sets $[\tau'_m, \tau'_M]$ and $[\tau''_m, \tau''_M]$ were considered. Fig. 7(b) shows a larger
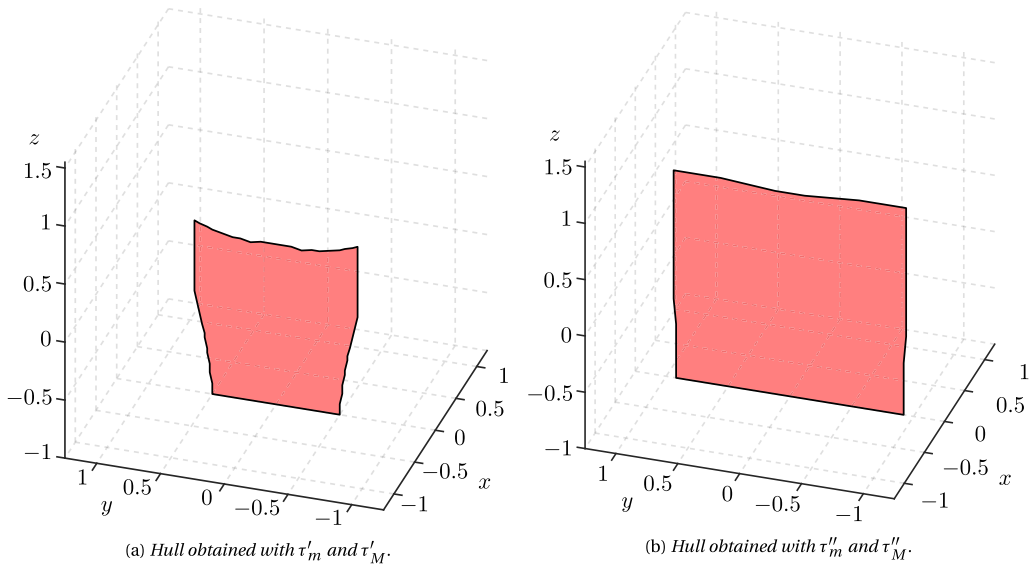
(a) *Hull obtained with τ'_m and τ'_M.*   (b) *Hull obtained with τ''_m and τ''_M.*

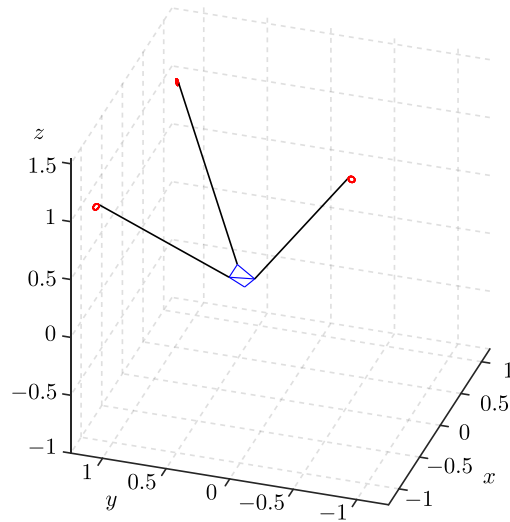**Fig. 5.** Hull representation of the 2-cable *UACDPR* workspace (units are in meters).

Fig. 6. Schematics of the 3-cable *UACDPR* for $\mathbf{p} = [0, 0.3, 0]^T$ (units are in meters). The *EE* is schematized in blue, cables are in black, and swiveling pulleys are in red. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 2
3-cable *UACDPR* geometric parameters.

i	1	2	3
$\mathbf{d}_i$ [m]	$\begin{bmatrix} -1 \\ 1.155 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 1.155 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 0 \\ -0.577 \\ 1 \end{bmatrix}$
r_i [m]	0.025	0.025	0.025
${}^P\mathbf{a}'_i$ [m]	$\begin{bmatrix} -0.1 \\ 0.116 \\ 0.1 \end{bmatrix}$	$\begin{bmatrix} 0.1 \\ 0.116 \\ 0.1 \end{bmatrix}$	$\begin{bmatrix} 0 \\ -0.058 \\ 0.1 \end{bmatrix}$

workspace than the one in Fig. 7(a) since the most limiting factors defining workspace boundaries appear to be the tension limits. It is worth observing that the considered 3-cable *UACDPR*, which is suspended, can reach most of the volume below its pulleys. The reachable volume laterally shrinks the more the *EE* is distant from its pulleys in the z direction, due to some cables reaching lower

(a) *Hull obtained with τ_m' and τ_M'.*

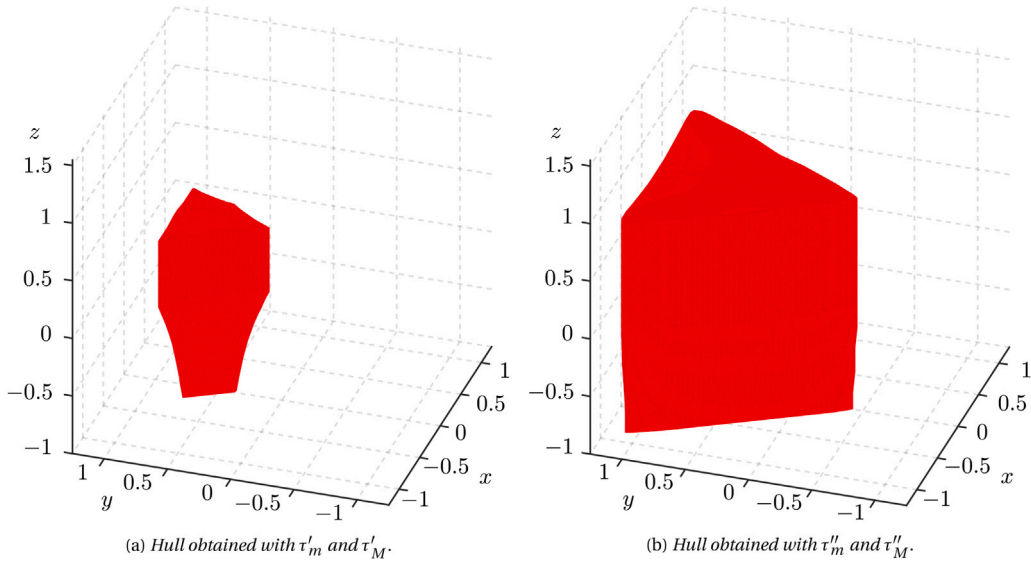(b) *Hull obtained with τ_m'' and τ_M''.*

Fig. 7. Hull representation of the 3-cable *UACDPR* workspace (units are in meters).

tension limits. The computational time was 3.1 min for the workspace in Fig. 7(b), and 15.3 min for the workspace in Fig. 7(a). In order to assess how much the analytical formulation of the Jacobian matrix impacts the computational time, the workspace in Fig. 7(b) was also obtained by solving the modified inverse *GSP* with *Matlab Fsolve* function, using a Levenberg–Marquardt algorithm, and sharing the workload related to finite-difference Jacobian approximation among 8 cores. Instead of 3.1 min, the computation time was 2 h and 22 min.

4.3. 4-cable UACDPRs

We propose a formulation for $\lambda \in \mathbb{R}$ that aims at reaching the largest wrench-feasible workspace possible, by looking for a tension distribution that is "the farthest" from the tension limits; our technique is derived from the *barycentric* approach of [41]. Due to Eq. (9), the wrench feasibility condition can be written as:

$$\tau_m \le \tau_0 + \tau^\perp \le \tau_M \tag{21}$$

Each *i*-th inequality ($i = 1, \dots, 4$) in Eq. (21) can be written as:

$$\tau_m - \tau_{0,i} \le \lambda \Xi_{F,i}^{T\perp} \le \tau_M - \tau_{0,i} \tag{22}$$

namely

$$\lambda_{m,i} \le \lambda \le \lambda_{M,i} \tag{23}$$

where

$$\lambda_{m,i} = \begin{cases} \left(\tau_m - \tau_{0,i}\right)/\Xi_{F,i}^{T\perp}, & \text{if } \Xi_{F,i}^{T\perp} \ge 0 \\ \left(\tau_M - \tau_{0,i}\right)/\Xi_{F,i}^{T\perp}, & \text{if } \Xi_{F,i}^{T\perp} < 0 \end{cases} \tag{24}$$

$$\lambda_{M,i} = \begin{cases} \left(\tau_M - \tau_{0,i}\right)/\Xi_{F,i}^{T\perp}, & \text{if } \Xi_{F,i}^{T\perp} \ge 0 \\ \left(\tau_m - \tau_{0,i}\right)/\Xi_{F,i}^{T\perp}, & \text{if } \Xi_{F,i}^{T\perp} < 0 \end{cases} \tag{25}$$

For each *EE* configuration, a wrench-feasible tension distribution is obtained if $\lambda_{m,max} = \max(\lambda_{m,1}, \dots, \lambda_{m,4}) \le \lambda_{M,min} = \min(\lambda_{M,1}, \dots, \lambda_{M,4})$. In case a feasible distribution exists, "the farthest" distribution from the tension limits is obtained as $\lambda = (\lambda_{m,max} + \lambda_{M,min})/2$. If a feasible distribution does not exist, the same formulation of λ results in the closest distribution to a feasible one.

The geometric parameters and a schematic of an exemplary 4-cable robot can be found in Table 3 and Fig. 8. The results of the workspace computation are shown in Fig. 9, where two tension limit sets $[\tau_m', \tau_M']$ and $[\tau_m'', \tau_M'']$ are considered. As for the 3-cable robot, Fig. 9(b) shows a larger workspace than the one in Fig. 9(a), since the most limiting factors defining workspace boundaries appear to be the tension limits. Like the 3-cable robot, the 4-cable *UACDPR*, in suspended configuration, can reach most of the volume below its pulleys. Computational time was 2.7 min for the workspace in Fig. 9(b), and 33.2 min for the workspace in Fig. 9(a).

We also investigate the workspace of a particular type of 4-cable robot, equivalent to a 3-cable robot with an additional cable installed in a more elevated position and whose exit point from the frame is centered in the xy plane with respect to the other three
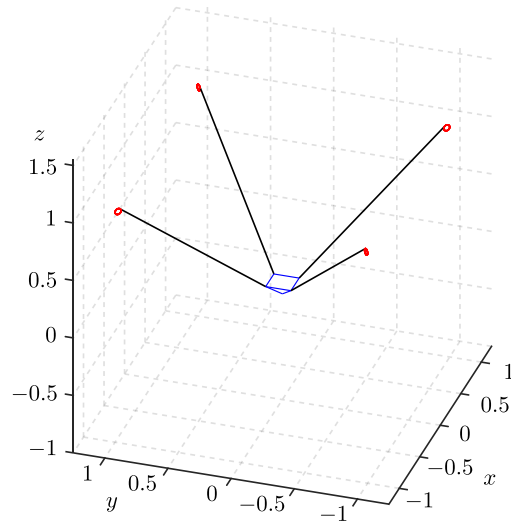
Fig. 8. Schematics of the 4-cable *UACDPR* for $\mathbf{p} = [0,0,0]^T$ (units are in meters). The *EE* is schematized in blue, cables are in black, and swiveling pulleys are in red. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

(a) *Hull obtained with τ'_m and τ'_M.*

(b) *Hull obtained with τ''_m and τ''_M.*
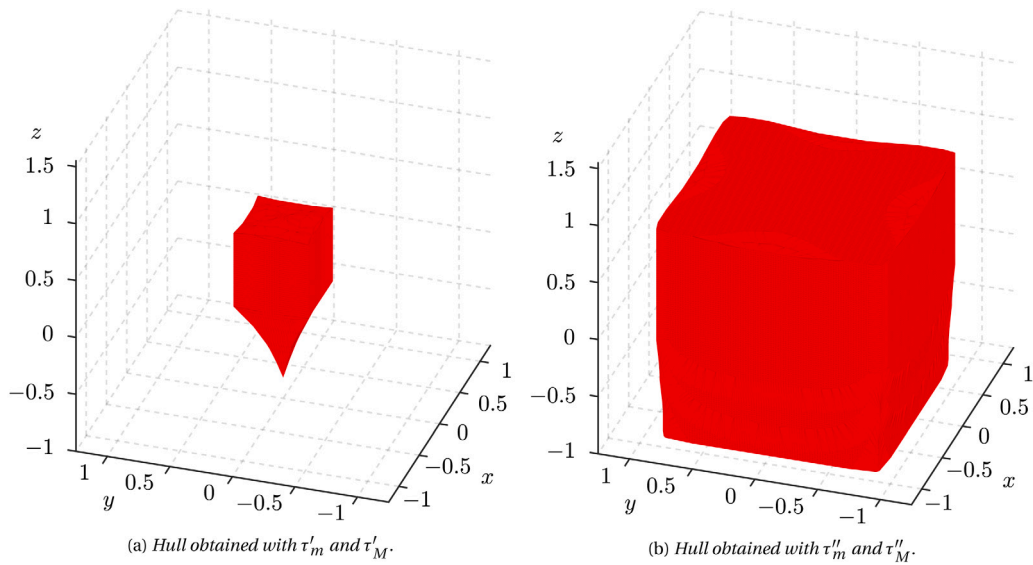
Fig. 9. Hull representation of the 4-cable *UACDPR* workspace (units are in meters).

Table 3
4-cable *UACDPR* geometric parameters.

i	1	2	3	4
$\mathbf{d}_i$ [m]	$\begin{bmatrix} -1 \\ 1 \\ 1 \end{bmatrix}$	$\begin{bmatrix} -1 \\ -1 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$
r_i [m]	0.025	0.025	0.025	0.025
$^P\mathbf{a}'_i$ [m]	$\begin{bmatrix} -0.1 \\ 0.1 \\ 0.1 \end{bmatrix}$	$\begin{bmatrix} -0.1 \\ -0.1 \\ 0.1 \end{bmatrix}$	$\begin{bmatrix} 0.1 \\ -0.1 \\ 0.1 \end{bmatrix}$	$\begin{bmatrix} 0.1 \\ 0.1 \\ 0.1 \end{bmatrix}$

Fig. 10. Schematics of the (3+1)-cable *UACDPR* for $\mathbf{p} = [0, 0.3, 0.2]^T$ (units are in meters). The *EE* is schematized in blue, cables are in black, and swiveling pulleys are in red. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)
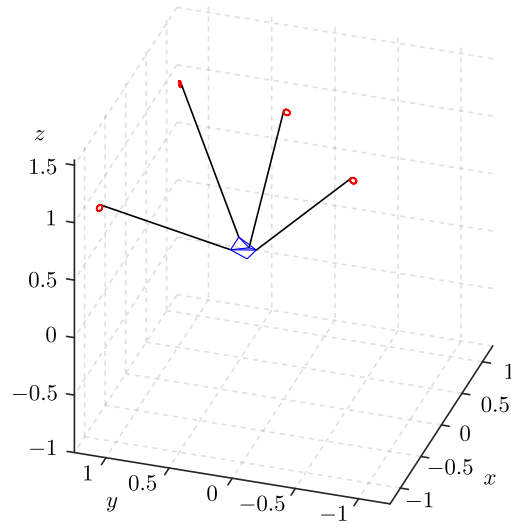

(a) *Hull obtained with τ'_m and τ'_M.*

(b) *Hull obtained with τ''_m and τ''_M.*
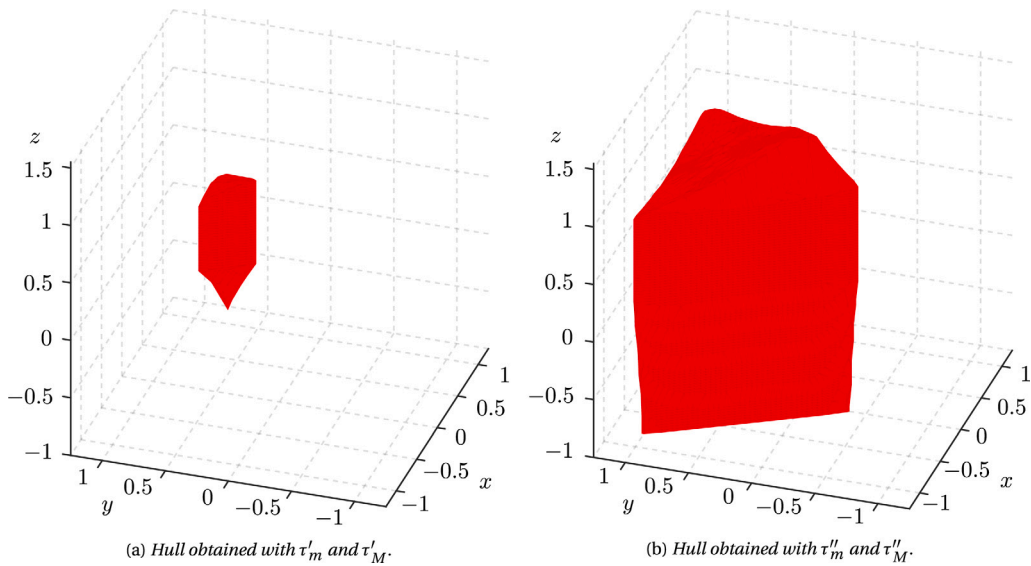
Fig. 11. Hull representation of the (3+1)-cable *UACDPR* workspace (units are in meters).

Table 4
(3+1)-cable *UACDPR* geometric parameters.

i	1	2	3	4
$\mathbf{d}_i$ [m]	$\begin{bmatrix} -1 \\ 1.155 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 1.155 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 0 \\ -0.05 \\ 1.5 \end{bmatrix}$	$\begin{bmatrix} 0 \\ -0.577 \\ 1 \end{bmatrix}$
r_i [m]	0.025	0.025	0.025	0.025
$^P\mathbf{a}'_i$ [m]	$\begin{bmatrix} -0.1 \\ 0.116 \\ 0.1 \end{bmatrix}$	$\begin{bmatrix} 0.1 \\ 0.116 \\ 0.1 \end{bmatrix}$	$\begin{bmatrix} 0 \\ 0 \\ 0.1 \end{bmatrix}$	$\begin{bmatrix} 0 \\ -0.058 \\ 0.1 \end{bmatrix}$

Fig. 12. Schematics of the (4+1)-cable *UACDPR* for $\mathbf{p} = [0, 0, 0.2]^T$ (units are in meters). The *EE* is schematized in blue, cables are in black, and swiveling pulleys are in red. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)
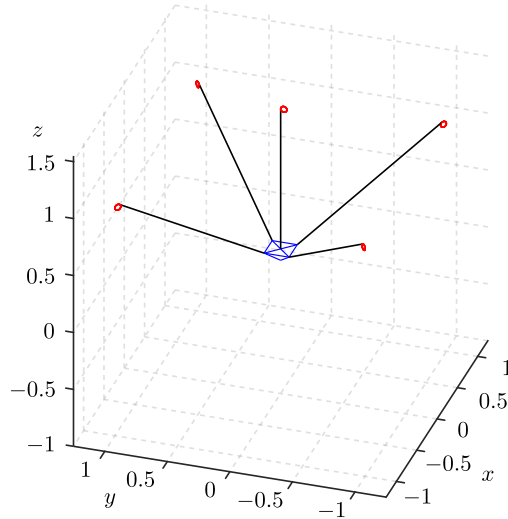
cables. Due to its peculiar geometry, we call this robot a (3+1)-cable *UACDPR*, and its geometric parameters and schematics can be found in Table 4 and Fig. 10. The results of the workspace computation are shown in Fig. 11, where two tension limit sets $[\tau'_m, \tau'_M]$ and $[\tau''_m, \tau''_M]$ are considered. It is interesting to notice that when narrow tension limits are considered (Fig. 11(a)), the workspace is smaller than the one of the 3-cable robot (Fig. 7(a)). On the contrary, for wider tension limits (Fig. 11(b)), the (3+1)-cable *UACDPR* can cover almost all the workspace of the 3-cable robot and is additionally able to reach the volume above the three lateral cables. Computational time was 1.9 min for the workspace in Fig. 11(b), and 14.5 min for the workspace in Fig. 11(a).

4.4. 5-cable UACDPR

For 5-cable *UACDPRs*, we must define a suitable formulation for $\lambda = [\lambda_1, \lambda_2]^T \in \mathbb{R}^2$. We adapted the method proposed in [45], which primarily aims at determining if a tension distribution exists by computing a *feasible* convex polygon in the space spanned by λ_1 and λ_2. Potential polygon vertices in the $\lambda_1 \lambda_2$ plane are determined by the intersections between the lines given by Eq. (9), which provide limit values for λ. Each vertex may be computed by considering the intersection between two lines in the $\lambda_1 \lambda_2$ plane, namely computing the solution to:

$$\begin{cases} \mathbf{\Xi}_{F,i}^{T\perp} \lambda & = \tau_u - \tau_{0,i} \\ \mathbf{\Xi}_{F,j}^{T\perp} \lambda & = \tau_v - \tau_{0,j} \end{cases} \tag{26}$$

where $u, v = m, M$ and $i, j = 1, \ldots, 5$, with $i \neq j$, and $\mathbf{\Xi}_{F,i}^{T\perp}$ and $\mathbf{\Xi}_{F,j}^{T\perp}$ are the i-th and j-th rows of $\mathbf{\Xi}_F^{T\perp}$, respectively. If a feasible polygon exists, it contains the λ's for which an admissible tension distribution can be computed. Without detailing how the polygon vertices can be efficiently found (see [45]), two cases are possible, depending on whether the polygon exists. If the polygon exists and has n_p vertices λ_q, with $q = 1, \ldots, n_p$, λ is computed as the algebraic mean of their coordinates:

$$\lambda = \frac{1}{n_p} \sum_{q=1}^{n_p} \lambda_q \tag{27}$$

If the polygon does not exist, λ is the vertex resulting in a tension distribution closest to wrench feasibility in the Euclidean sense, as in [46]. Analytically speaking, λ is still described by Eq. (27), but with $n_p = 1$, and with a different choice of λ_q.

As a 5-cable *UACDPR*, we propose a particular architecture investigated in [47], which is equivalent to a 4-cable robot with an additional cable installed in a more elevated position and whose exit point from the frame is centered in the xy plane with respect to the other four cables. We call this robot a (4+1)-cable *UACDPR*, and its geometric parameters and schematics can be found in Table 5 and Fig. 12. The results of the workspace computation are shown in Fig. 13, where two tension limit sets $[\tau'_m, \tau'_M]$ and $[\tau''_m, \tau''_M]$ are considered. Similarly to the comparison between the (3+1)-cable robot and the 3-cable one, when narrow tension limits are considered (Fig. 13(a)), the workspace volume of the (4+1)-cable *UACDPR* is smaller than the one of the 4-cable robot (Fig. 9(a)). On the contrary, for wider tension limits (Fig. 13(b)), the (4+1)-cable *UACDPR* is able to cover almost all the workspace of the 4-cable robot and is additionally able to reach the volume above the four lateral cables. Computational time was 14.5 min for the workspace in Fig. 13(b), and 53.9 min for the workspace in Fig. 13(a). The $(n+1)$-cable robot architectures, with $n = 3, 4$, appear to be useful, if suitable tension limits are used, for tasks involving the positioning of a payload in the same workspace of n-cable

Table 5
(4+1)-cable *UACDPR* geometric parameters.

i	1	2	3	4	5
$\mathbf{d}_i$ [m]	$\begin{bmatrix} -1 \\ 1 \\ 1 \end{bmatrix}$	$\begin{bmatrix} -1 \\ -1 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 0 \\ -0.05 \\ 1.5 \end{bmatrix}$	$\begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$
r_i [m]	0.025	0.025	0.025	0.025	0.025
$^P\mathbf{a}_i'$ [m]	$\begin{bmatrix} -0.1 \\ 0.1 \\ 0.1 \end{bmatrix}$	$\begin{bmatrix} -0.1 \\ -0.1 \\ 0.1 \end{bmatrix}$	$\begin{bmatrix} 0 \\ 0 \\ 0.1 \end{bmatrix}$	$\begin{bmatrix} 0.1 \\ -0.1 \\ 0.1 \end{bmatrix}$	$\begin{bmatrix} 0.1 \\ 0.1 \\ 0.1 \end{bmatrix}$

(a) *Hull obtained with τ_m' and τ_M'.*   (b) *Hull obtained with τ_m'' and τ_M''.*
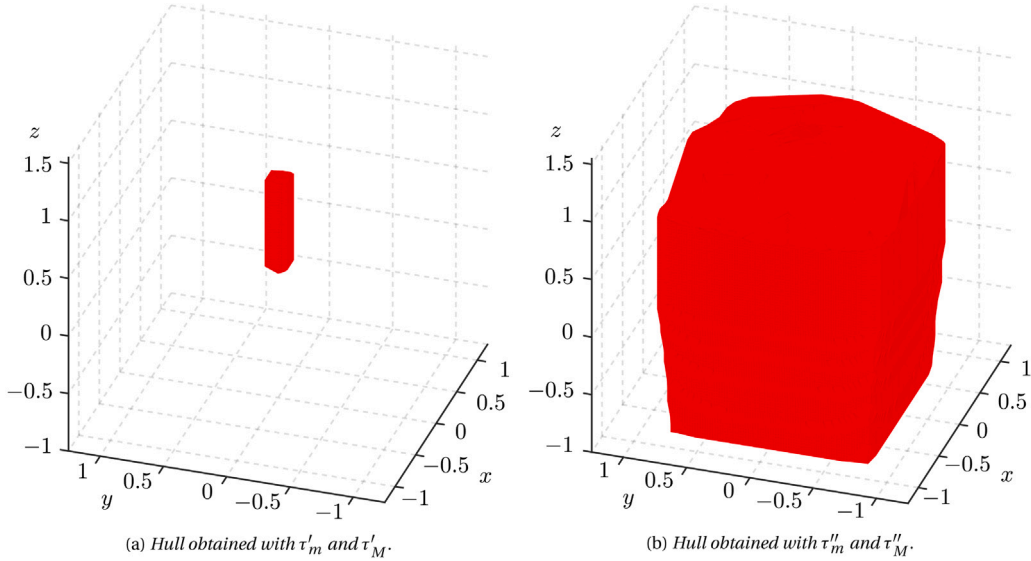
Fig. 13. Hull representation of the (4+1)-cable *UACDPR* workspace (units are in meters).

robots, but also requiring to lift such load above the installation area of the n-cable robot pulleys, as the one presented in [47]. Depending on n, the overall rough shape of the workspace changes, and one can use $n = 3$ if a triangular prism workspace is needed or $n = 4$ if a rectangular prism is more appropriate.

5. Application: Trajectory planning

This section (i) highlights the merits of our methodology for the computation of statically feasible and continuous trajectory set points, and (ii) shows that our method is capable of correctly computing the workspace boundaries.

A series of successive set points is computed by solving Eq. (11) along four linear trajectories, with each one of them connecting two points on the plane characterized by $z = -0.5$ m in the 4-cable *UACDPR* workspace shown in Fig. 9(b) (see Fig. 14(a)). The first trajectory starts from a point A well inside the workspace borders (the cable tension limits considered in this example are $\tau_m'' = 5$ N and $\tau_M'' = 500$ N) and arrives in B, on the workspace border as computed in Section 4.3. The second and third trajectories, from B to C and from C to D respectively, are outside the workspace border due to some cable tensions being below the minimum threshold τ_m''.[4] Once the *EE* reaches D, on the border, its returns with a final trajectory to A. Each assigned position set point is reported in Fig. 14(b), and, computed as:

$$\boldsymbol{p}_{ij}(s_k) = \boldsymbol{p}_i + \frac{(\boldsymbol{p}_j - \boldsymbol{p}_i)s_k}{\|\boldsymbol{p}_j - \boldsymbol{p}_i\|}, \qquad i = A,\ B,\ C,\ D \qquad j = B,\ C,\ D,\ A \tag{28}$$

where:

$$\boldsymbol{p}_A = \begin{bmatrix} 0.6 \\ 0.6 \\ -0.5 \end{bmatrix} \text{m}, \qquad \boldsymbol{p}_B = \begin{bmatrix} 0.7 \\ 0.9 \\ -0.5 \end{bmatrix} \text{m}, \qquad \boldsymbol{p}_C = \begin{bmatrix} 0.9 \\ 0.9 \\ -0.5 \end{bmatrix} \text{m}, \qquad \boldsymbol{p}_D = \begin{bmatrix} 0.9 \\ 0.7 \\ -0.5 \end{bmatrix} \text{m}$$

---

[4] Eq. (11) is solved with a tension distribution which is the nearest to a feasible one, as introduced in Section 4.3.
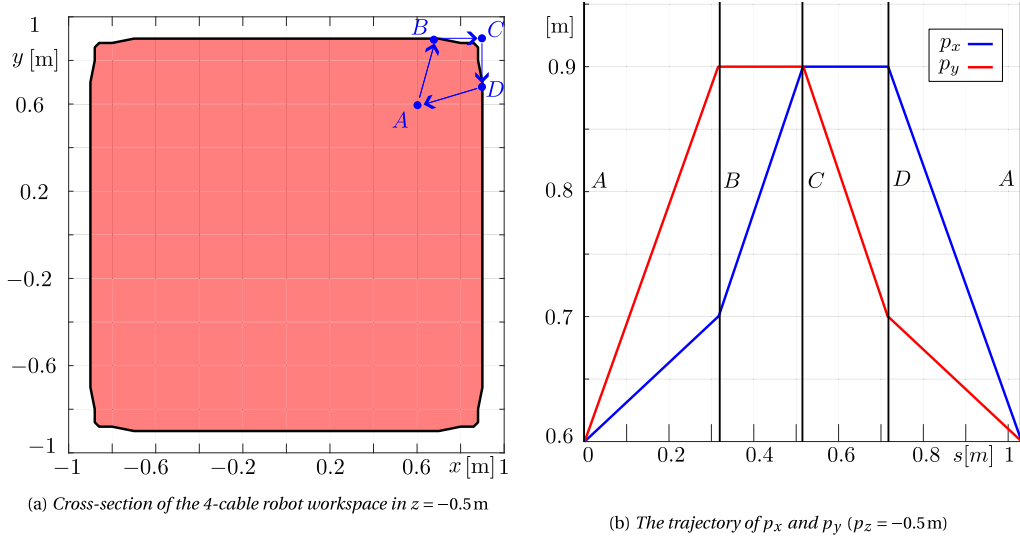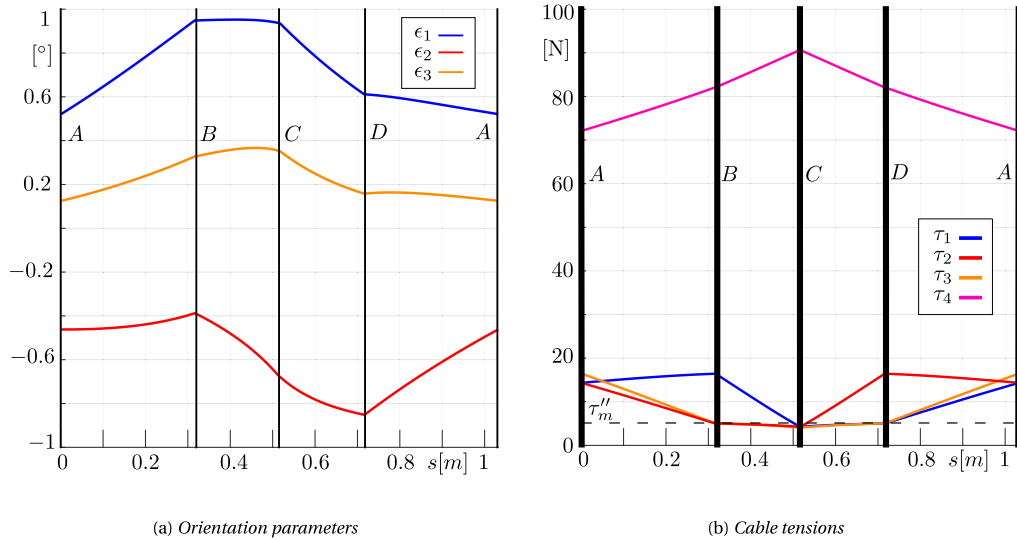
(a) *Cross-section of the 4-cable robot workspace in $z = -0.5$ m*

(b) *The trajectory of p_x and p_y ($p_z = -0.5$ m)*

Fig. 14. Assigned trajectories.

(a) *Orientation parameters*

(b) *Cable tensions*

Fig. 15. The modified inverse *GSP* results over the assigned trajectories.

$$s_k \in 0, 0.001, 0.002, \ldots, \|\boldsymbol{p}_j - \boldsymbol{p}_i\|$$

In Fig. 14(b), the abscissa s cumulates the total distance of each s_k. The results of the modified inverse *GSP* over the assigned trajectories are shown in Fig. 15. The computed orientation parameters ϵ are displayed in Fig. 15(a), where the continuity of each problem solution can be appreciated. Fig. 15(b) shows the cable tensions computed after the problem solution. It can be seen that the tensions of several cables fall below τ_m'' when the *EE* set point is between point B and C, or C and D, as expected. Nonetheless, the computed tensions are continuous.

6. Conclusions

This paper introduced an algorithm for the computation of the reachable static workspace of generic *UACDPRs* and demonstrated its application on robots with 3, 4, and 5 cables. The algorithm was based on the definition of a modified inverse geometrico-static problem, which allows one to compute feasible *EE* orientations characterized by a desired tension distribution when the *EE* position is assigned. An analytical formulation of the Jacobian of the modified inverse geometrico-static problem was also introduced to speed up computation. The application of the algorithm showed that, for any architecture, the most limiting factors for the reachable

static workspace were the limits set for the cable tensions. Additionally, we showed that our algorithm is able to compute continuous statically feasible set points to be used for trajectory planning.

Declaration of competing interest

Data availability

Data will be made available on request.

Acknowledgments

Appendix. Analytical formulation of the Jacobian of the modified inverse *GSP*

Even though $\boldsymbol{J}$ is the derivative of $\boldsymbol{\phi}$ with respect to $\boldsymbol{\epsilon}$ (see Eq. (11)), it is convenient to differentiate $\boldsymbol{J}$ with respect to the full *EE* pose $\boldsymbol{\zeta}$, because this leads to a simpler analytical expression. We start by differentiating the left side Eq. (4), which yields [4]:

$$\frac{\partial \left(\boldsymbol{\Xi}^T \boldsymbol{\tau} - \boldsymbol{f} \right)}{\partial \boldsymbol{\zeta}} = \frac{\partial \boldsymbol{\Xi}^T}{\partial \boldsymbol{\zeta}} \boldsymbol{\tau} - \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{\zeta}} + \boldsymbol{\Xi}^T \frac{\partial \boldsymbol{\tau}}{\partial \boldsymbol{\zeta}} \tag{A.1}$$

where:

$$\frac{\partial \boldsymbol{\Xi}^T}{\partial \boldsymbol{\zeta}} \boldsymbol{\tau} \overset{\Delta}{=} \mathbf{KD}, \qquad -\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{\zeta}} \overset{\Delta}{=} \mathbf{QD} \tag{A.2}$$

$\mathbf{K}$ and $\mathbf{Q}$ are respectively given in Eqs. (15) and (16). Matrix $\mathbf{D} \in \mathbb{R}^{6 \times 6}$, relating the *EE* twist $\boldsymbol{v}$ with $\dot{\boldsymbol{\zeta}}$, only depends on the choice of the orientation parametrization; for example, if $\boldsymbol{\epsilon} = \left[\theta_x, \theta_y, \theta_y \right]^T$, and $\mathbf{R} = \boldsymbol{R}_x \left(\theta_x \right) \boldsymbol{R}_y \left(\theta_y \right) \boldsymbol{R}_z \left(\theta_z \right)$, with $\boldsymbol{R}_x$, $\boldsymbol{R}_y$, and $\boldsymbol{R}_z$ being elementary rotation matrices about x, y, and z axes, and $c_x = \cos x$ and $s_x = \sin x$, we have [12]:

$$\boldsymbol{v} = \mathbf{D}(\boldsymbol{\epsilon}) \dot{\boldsymbol{\zeta}}, \qquad \mathbf{D}(\boldsymbol{\epsilon}) \overset{\Delta}{=} \begin{bmatrix} \mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \mathbf{H}(\boldsymbol{\epsilon}) \end{bmatrix}, \qquad \mathbf{H}(\boldsymbol{\epsilon}) = \begin{bmatrix} 1 & 0 & s_{\theta_y} \\ 0 & c_{\theta_x} & -s_{\theta_x} c_{\theta_y} \\ 0 & s_{\theta_x} & c_{\theta_x} c_{\theta_y} \end{bmatrix} \tag{A.3}$$

The leftmost definition in Eq. (A.2) is a product between the derivative of a matrix with respect to a vector, which is a 3-dimensional tensor, and a vector, and the result of this operation is a matrix. Computational details can be found in the Appendix of [4].

The expression of $\mathbf{J}$ is more easily derived by considering Eq. (8) instead of Eq. (11) since they only differ for the explicit expression of the tension distribution. Accordingly:

$$\mathbf{J} = \frac{\partial \left(\boldsymbol{\Xi}_M^T \boldsymbol{\tau} - m \bar{\boldsymbol{s}}' \boldsymbol{g} \right)}{\partial \boldsymbol{\zeta}} = \frac{\partial \boldsymbol{\Xi}_M^T}{\partial \boldsymbol{\zeta}} \boldsymbol{\tau} - m \frac{\partial \bar{\boldsymbol{s}}' \boldsymbol{g}}{\partial \boldsymbol{\zeta}} + \boldsymbol{\Xi}_M^T \frac{\partial \boldsymbol{\tau}}{\partial \boldsymbol{\zeta}} \tag{A.4}$$

The first two terms on the right side of Eq. (A.4) are simply the last three rows (namely rows from 4 to 6) of the first two terms on the right side of Eq. (A.1), which are provided in Eq. (A.2) and denoted by $\left(\mathbf{K}_{46} + \mathbf{Q}_{46} \right) \mathbf{D}$. The rest of the Jacobian is then computed by evaluating the derivative of the tension distribution $\boldsymbol{\tau}$ in Eq. (9) as:

$$\frac{\partial \boldsymbol{\tau}}{\partial \boldsymbol{\zeta}} = \frac{\partial \boldsymbol{\tau}_0}{\partial \boldsymbol{\zeta}} + \frac{\partial \boldsymbol{\tau}^{\perp}}{\partial \boldsymbol{\zeta}}, \qquad \frac{\partial \boldsymbol{\tau}_0}{\partial \boldsymbol{\zeta}} = m \frac{\partial \left(\boldsymbol{\Xi}_F^{-T} \right)}{\partial \boldsymbol{\zeta}} \boldsymbol{g}, \qquad \frac{\partial \boldsymbol{\tau}^{\perp}}{\partial \boldsymbol{\zeta}} = \frac{\partial \left(\boldsymbol{\Xi}_F^{T\perp} \right)}{\partial \boldsymbol{\zeta}} \boldsymbol{\lambda} + \boldsymbol{\Xi}_F^{\perp} \frac{\partial \boldsymbol{\lambda}}{\partial \boldsymbol{\zeta}} \tag{A.5}$$

where it is worth remarking that $\partial \boldsymbol{\tau}^{\perp} / \partial \boldsymbol{\zeta}$ is defined only for $n = 4$ or 5. The calculation of $\partial \boldsymbol{\lambda} / \partial \boldsymbol{\zeta}$ depends on the specific choice of tension distribution algorithm, and it will be detailed in Appendix A.1 for the 4-cable *UACDPR*, and in Appendix A.2 for the 5-cable *UACDPR*. The remaining derivatives in Eq. (A.5) depend on the specific formulations employed for $\boldsymbol{\Xi}_F^{-T}$ and $\boldsymbol{\Xi}_F^{\perp}$, since they are not unique. We propose using a formulation borrowed from [20], which simplifies computation. We partition the cable tension array $\boldsymbol{\tau}$ and matrix $\boldsymbol{\Xi}_F^T$ so that:

$$\boldsymbol{\Xi}_F^T \boldsymbol{\tau} = \boldsymbol{\Xi}_{Fd}^T \boldsymbol{\tau}_d + \boldsymbol{\Xi}_{Fc}^T \boldsymbol{\tau}_c, \qquad \boldsymbol{\tau} = \left[\boldsymbol{\tau}_d^T \ \boldsymbol{\tau}_c^T \right]^T, \qquad \boldsymbol{\Xi}_F^T = \left[\boldsymbol{\Xi}_{Fd} \ \boldsymbol{\Xi}_{Fc} \right]^T \tag{A.6}$$

where $\boldsymbol{\tau}_d \in \mathbb{R}^{3 \times 1}$ and $\boldsymbol{\tau}_c \in \mathbb{R}^{(3-\kappa) \times 1}$ are the first 3 and last $3 - \kappa$ elements of $\boldsymbol{\tau}$, and $\boldsymbol{\Xi}_{Fd} \in \mathbb{R}^{3 \times 3}$ and $\boldsymbol{\Xi}_{Fc} \in \mathbb{R}^{3 \times (3-\kappa)}$ are the first 3 and last $3 - \kappa$ columns of $\boldsymbol{\Xi}_F$, respectively. Ref. [20] showed that, according to this partition, $\boldsymbol{\lambda}$ holds a physical meaning, which is $\boldsymbol{\lambda} = \boldsymbol{\tau}_c$, if and only if:

$$\boldsymbol{\Xi}_F^{-T} = \begin{bmatrix} \boldsymbol{\Xi}_{Fd}^{-T} \\ \mathbf{0}_{\kappa \times 3} \end{bmatrix}, \qquad \boldsymbol{\Xi}_F^{T\perp} = \begin{bmatrix} -\boldsymbol{\Xi}_{Fd}^{-T} \boldsymbol{\Xi}_{Fc}^T \\ \mathbf{I}_{\kappa \times \kappa} \end{bmatrix} \tag{A.7}$$

and consequently, if Eq. (A.7) is substituted in Eqs. (7) and (9), it can be shown that:

$$\boldsymbol{\tau}_d = m\boldsymbol{\Xi}_{Fd}^{-T}\mathbf{g}, \qquad \boldsymbol{\tau}_0 = \begin{bmatrix} \boldsymbol{\tau}_d^T & \mathbf{0}_{1\times(3-\kappa)} \end{bmatrix}^T, \qquad \boldsymbol{\tau}^\perp = \boldsymbol{\Xi}_F^{T\perp}\boldsymbol{\tau}_c \tag{A.8}$$

Notice that $\boldsymbol{\Xi}_F^{-T} = \boldsymbol{\Xi}_{Fd}^{-T}$ for $n = 3$, and $\boldsymbol{\Xi}_{Fd}^T$ is not singular, thus invertible, if the first three cables are not coplanar, since the rows of $\boldsymbol{\Xi}_{Fd}^T$ are the cable directions $\boldsymbol{t}_i^T$, with $i = 1, 2, 3$. Accordingly, as long as at least three cables are not coplanar, one can always re-order cables to obtain a non-singular $\boldsymbol{\Xi}_{Fd}^T$. According to this formulation, we can evaluate $\partial\boldsymbol{\tau}_0/\partial\boldsymbol{\zeta}$ and the first term of $\partial\boldsymbol{\tau}^\perp/\partial\boldsymbol{\zeta}$ as[5]:

$$\frac{\partial\boldsymbol{\tau}_0}{\partial\boldsymbol{\zeta}} = \begin{bmatrix} m\frac{\partial\boldsymbol{\Xi}_{Fd}^{-T}}{\partial\boldsymbol{\zeta}}\mathbf{g} \\ \mathbf{0}_{\kappa\times 6} \end{bmatrix} = \begin{bmatrix} -m\boldsymbol{\Xi}_{Fd}^{-T}\frac{\partial\boldsymbol{\Xi}_{Fd}^T}{\partial\boldsymbol{\zeta}}\boldsymbol{\Xi}_{Fd}^{-T}\mathbf{g} \\ \mathbf{0}_{\kappa\times 6} \end{bmatrix} = \begin{bmatrix} -\boldsymbol{\Xi}_{Fd}^{-T}\frac{\partial\boldsymbol{\Xi}_{Fd}^T}{\partial\boldsymbol{\zeta}}\boldsymbol{\tau}_d \\ \mathbf{0}_{\kappa\times 6} \end{bmatrix} \tag{A.9}$$

$$\frac{\partial\left(\boldsymbol{\Xi}_F^{T\perp}\right)}{\partial\boldsymbol{\zeta}}\boldsymbol{\lambda} = -\begin{bmatrix} \left(\frac{\partial\boldsymbol{\Xi}_{Fd}^{-T}}{\partial\boldsymbol{\zeta}}\boldsymbol{\Xi}_{Fc}^T + \boldsymbol{\Xi}_{Fd}^{-T}\frac{\partial\boldsymbol{\Xi}_{Fc}^T}{\partial\boldsymbol{\zeta}}\right)\boldsymbol{\lambda} \\ \mathbf{0}_{\kappa\times 6} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\Xi}_{Fd}^{-T}\left(\frac{\partial\boldsymbol{\Xi}_{Fd}^T}{\partial\boldsymbol{\zeta}}\boldsymbol{\Xi}_{Fd}^{-T}\boldsymbol{\Xi}_{Fc}^T - \frac{\partial\boldsymbol{\Xi}_{Fc}^T}{\partial\boldsymbol{\zeta}}\mathbf{I}_{\kappa\times\kappa}\right)\boldsymbol{\lambda} \\ \mathbf{0}_{\kappa\times 6} \end{bmatrix} \tag{A.10}$$

Both Eq. (A.9) and (A.10) contain products between 3-dimensional tensors and vectors, as in Eq. (A.2). According to the technique presented in the Appendix of [4], it can be shown that:

$$\frac{\partial\boldsymbol{\Xi}_{Fd}^T}{\partial\boldsymbol{\zeta}}\boldsymbol{\tau}_d = \mathbf{K}_{Fd}\mathbf{D}, \qquad \mathbf{K}_{Fd} \overset{\Delta}{=} \sum_{i=1}^{3}\tau_{0,i}\begin{bmatrix} \mathbf{T}_i & -\mathbf{T}_i\tilde{\boldsymbol{a}}_i' \end{bmatrix} \tag{A.11}$$

$$\left(\frac{\partial\boldsymbol{\Xi}_{Fd}^T}{\partial\boldsymbol{\zeta}}\boldsymbol{\Xi}_{Fd}^{-T}\boldsymbol{\Xi}_{Fc}^T - \frac{\partial\boldsymbol{\Xi}_{Fc}^T}{\partial\boldsymbol{\zeta}}\mathbf{I}_{\kappa\times\kappa}\right)\boldsymbol{\lambda} = -\mathbf{K}_F^\perp\mathbf{D}, \qquad \mathbf{K}_F^\perp \overset{\Delta}{=} \sum_{i=1}^{n}\tau_i^\perp\begin{bmatrix} \mathbf{T}_i & -\mathbf{T}_i\tilde{\boldsymbol{a}}_i' \end{bmatrix} \tag{A.12}$$

Thus substituting Eq. (A.11) in Eq. (A.9), and Eq. (A.12) in Eq. (A.10):

$$\frac{\partial\boldsymbol{\tau}_0}{\partial\boldsymbol{\zeta}} = -\begin{bmatrix} \boldsymbol{\Xi}_{Fd}^{-T}\mathbf{K}_{Fd} \\ \mathbf{0}_{\kappa\times 6} \end{bmatrix}\mathbf{D}, \qquad \frac{\partial\left(\boldsymbol{\Xi}_F^{T\perp}\right)}{\partial\boldsymbol{\zeta}}\boldsymbol{\lambda} = -\begin{bmatrix} \boldsymbol{\Xi}_{Fd}^{-T}\mathbf{K}_F^\perp \\ \mathbf{0}_{\kappa\times 6} \end{bmatrix}\mathbf{D} \tag{A.13}$$

Once $\partial\boldsymbol{\lambda}/\partial\boldsymbol{\zeta}$ is also computed, the Jacobian $\mathbf{J}$ is given by the last 3 columns of Eq. (A.4), namely the last 3 columns of:

$$\left(\mathbf{K}_{46} + \mathbf{Q}_{46} - \boldsymbol{\Xi}_M^T\left(\begin{bmatrix} \boldsymbol{\Xi}_{Fd}^{-T}\mathbf{K}_{Fd} \\ \mathbf{0}_{\kappa\times 6} \end{bmatrix} + \begin{bmatrix} \boldsymbol{\Xi}_{Fd}^{-T}\mathbf{K}_F^\perp \\ \mathbf{0}_{\kappa\times 6} \end{bmatrix}\right)\right)\mathbf{D} + \boldsymbol{\Xi}_M^T\boldsymbol{\Xi}_F^{T\perp}\frac{\partial\boldsymbol{\lambda}}{\partial\boldsymbol{\zeta}} \tag{A.14}$$

.

A.1. Computation of $\partial\boldsymbol{\lambda}/\partial\boldsymbol{\zeta}$ for the 4-cable UACDPR

In this case (see Section 4.3) λ is a scalar and equal to $\lambda = (\lambda_{m,max} + \lambda_{M,min})/2$, where $\lambda_{m,max} = \max(\lambda_{m,1}, \ldots, \lambda_{m,4})$, $\lambda_{M,min} = \min(\lambda_{M,1}, \ldots, \lambda_{M,4})$, and $\lambda_{m,i}$ and $\lambda_{M,i}$ for $i = 1, \ldots, n$ are given by Eqs. (24) and (25) respectively. Accordingly:

$$\frac{\partial\lambda}{\partial\boldsymbol{\zeta}} = \frac{1}{2}\left(\frac{\partial\lambda_{m,max}}{\partial\boldsymbol{\zeta}} + \frac{\partial\lambda_{M,min}}{\partial\boldsymbol{\zeta}}\right) \tag{A.15}$$

where, if the cable indexes maximizing λ_m and minimizing λ_M are respectively denoted by α and β:

$$\frac{\partial\lambda_{m,max}}{\partial\boldsymbol{\zeta}} = \begin{cases} \dfrac{-\boldsymbol{\Xi}_{F,\alpha}^{T\perp}\partial\tau_{0,\alpha}/\partial\boldsymbol{\zeta} - (\tau_m - \tau_{0,\alpha})\partial\boldsymbol{\Xi}_{F,\alpha}^{T\perp}/\partial\boldsymbol{\zeta}}{\left(\boldsymbol{\Xi}_{F,\alpha}^{T\perp}\right)^2}, & \text{if } \boldsymbol{\Xi}_{F,\alpha}^{T\perp} \geq 0 \\[3mm] \dfrac{-\boldsymbol{\Xi}_{F,\alpha}^{T\perp}\partial\tau_{0,\alpha}/\partial\boldsymbol{\zeta} - (\tau_M - \tau_{0,\alpha})\partial\boldsymbol{\Xi}_{F,\alpha}^{T\perp}/\partial\boldsymbol{\zeta}}{\left(\boldsymbol{\Xi}_{F,\alpha}^{T\perp}\right)^2}, & \text{if } \boldsymbol{\Xi}_{F,\alpha}^{T\perp} < 0 \end{cases} \tag{A.16}$$

$$\frac{\partial\lambda_{M,min}}{\partial\boldsymbol{\zeta}} = \begin{cases} \dfrac{-\boldsymbol{\Xi}_{F,\beta}^{T\perp}\partial\tau_{0,\beta}/\partial\boldsymbol{\zeta} - (\tau_M - \tau_{0,\beta})\partial\boldsymbol{\Xi}_{F,\beta}^{T\perp}/\partial\boldsymbol{\zeta}}{\left(\boldsymbol{\Xi}_{F,\beta}^{T\perp}\right)^2}, & \text{if } \boldsymbol{\Xi}_{F,\beta}^{T\perp} \geq 0 \\[3mm] \dfrac{-\boldsymbol{\Xi}_{F,\beta}^{T\perp}\partial\tau_{0,\beta}/\partial\boldsymbol{\zeta} - (\tau_m - \tau_{0,\beta})\partial\boldsymbol{\Xi}_{F,\beta}^{T\perp}/\partial\boldsymbol{\zeta}}{\left(\boldsymbol{\Xi}_{F,\beta}^{T\perp}\right)^2}, & \text{if } \boldsymbol{\Xi}_{F,\beta}^{T\perp} < 0 \end{cases} \tag{A.17}$$

The derivative of $\tau_{0,i}$ with respect to $\boldsymbol{\zeta}$ is provided by the i-th row of the leftmost term in Eq. (A.13), whereas $\partial\boldsymbol{\Xi}_{F,\alpha}^{T\perp}/\partial\boldsymbol{\zeta}$ and $\partial\boldsymbol{\Xi}_{F,\beta}^{T\perp}/\partial\boldsymbol{\zeta}$ can be shown to be, respectively, the α-th and the β-th row of:

$$\frac{\partial\boldsymbol{\Xi}_F^{T\perp}}{\partial\boldsymbol{\zeta}} = -\begin{bmatrix} \boldsymbol{\Xi}_{Fd}^{-T}\mathbf{A} \\ \mathbf{0}_{1\times 6} \end{bmatrix}\mathbf{D}, \qquad \mathbf{A} \overset{\Delta}{=} \sum_{i=1}^{4}\boldsymbol{\Xi}_{F,i}^{T\perp}\begin{bmatrix} \mathbf{T}_i & -\mathbf{T}_i\tilde{\boldsymbol{a}}_i' \end{bmatrix} \tag{A.18}$$

---

[5] We recall that, since $\boldsymbol{\Xi}_{Fd}^T$ is an invertible matrix, then $\boldsymbol{\Xi}_{Fd}^T\boldsymbol{\Xi}_{Fd}^{-T} = \mathbf{I}_{3\times 3}$ and $\boldsymbol{\Xi}_{Fd}^T\boldsymbol{\Xi}_{Fd}^{-T}\boldsymbol{v} = \boldsymbol{v}$ for any $\boldsymbol{v} \in \mathbb{R}^{3\times 1}$. If $\boldsymbol{v}$ does not depend on $\boldsymbol{\zeta}$, then $\partial\boldsymbol{\Xi}_{Fd}^T/\partial\boldsymbol{\zeta}\,\boldsymbol{\Xi}_{Fd}^{-T}\boldsymbol{v} + \boldsymbol{\Xi}_{Fd}^T\partial\boldsymbol{\Xi}_{Fd}^{-T}/\partial\boldsymbol{\zeta}\,\boldsymbol{v} = \mathbf{0}_{3\times 6}$, and thus $\partial\boldsymbol{\Xi}_{Fd}^{-T}/\partial\boldsymbol{\zeta}\,\boldsymbol{v} = -\boldsymbol{\Xi}_{Fd}^{-T}\partial\boldsymbol{\Xi}_{Fd}^T/\partial\boldsymbol{\zeta}\,\boldsymbol{\Xi}_{Fd}^{-T}\boldsymbol{v}$.

A.2. Computation of $\partial\lambda/\partial\zeta$ for the 5-cable UACDPR

For the 5-cable *UACDPR*, λ is computed in Eq. (27) as the algebraic means of points λ_q in the $\lambda_1\lambda_2$ plane, computed in their turn as the solution of Eq. (26). As a consequence:

$$\frac{\partial\lambda}{\partial\zeta} = \frac{1}{n_p}\sum_{q=1}^{n_p}\frac{\partial\lambda_q}{\partial\zeta} \tag{A.19}$$

where $\partial\lambda_q/\partial\zeta$ can be computed, after appropriately selecting i,j,u and v, by differentiating Eq. (26). If λ_q is the solution of Eq. (26) for assigned i,j,u and v, and $\tau_q^\perp = \Xi_F^{T\perp}\lambda_q$, then:

$$\frac{\partial\lambda_q}{\partial\zeta} = -\Xi_{F,ij}^{-T\perp}\left(\frac{\partial\tau_{0,ij}}{\partial\zeta} + \frac{\partial\Xi_{F,ij}^{T\perp}}{\partial\zeta}\lambda_q\right), \qquad \Xi_{F,ij}^{T\perp} = \begin{bmatrix}\Xi_{F,i}^{T\perp}\\\Xi_{F,j}^{T\perp}\end{bmatrix} \tag{A.20}$$

where $\partial\tau_{0,ij}/\partial\zeta$ is a matrix composed by the i-th and j-th rows of the leftmost term in Eq. (A.13), and $\left(\partial\Xi_{F,ij}^{T\perp}/\partial\zeta\right)\lambda_q$ can be shown to be the matrix composed by the i-th and j-th rows of:

$$\frac{\partial\Xi_F^{T\perp}}{\partial\zeta}\lambda_q = -\begin{bmatrix}\Xi_{Fd}^{-T}\mathbf{K}_q^\perp\\\mathbf{0}_{2\times6}\end{bmatrix}\mathbf{D}, \qquad \mathbf{K}_q^\perp \overset{\Delta}{=} \sum_{i=1}^{5}\tau_{q,i}^\perp\begin{bmatrix}\mathbf{T}_i & -\mathbf{T}_i\tilde{a}_i'\end{bmatrix} \tag{A.21}$$

References
[1] P. Miermeister, M. Lächele, R. Boss, C. Masone, C. Schenk, J. Tesch, M. Kerger, H. Teufel, A. Pott, 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, 2016, pp. 3024–3029.
[2] G. Abbasnejad, M. Carricato, Direct geometrico-static problem of underconstrained cable-driven parallel robots with n cables, IEEE Trans. Robot. 31 (2) (2015) 468–478.
[3] S.W. Hwang, J.-H. Bak, J. Yoon, J.H. Park, Oscillation reduction and frequency analysis of under-constrained cable-driven parallel robot with three cables, Robotica 38 (3) (2020) 375–395.
[4] E. Idà, S. Briot, M. Carricato, Natural oscillations of underactuated cable-driven parallel robots, IEEE Access 9 (2021) 71660–71672.
[5] M. Conconi, M. Carricato, A new assessment of singularities of parallel kinematic chains, IEEE Trans. Robot. 25 (4) (2009) 757–770.
[6] S. Schröder, Under constrained cable-driven parallel robot for vertical green maintenance, in: M. Gouttefarde, T. Bruckmann, A. Pott (Eds.), Cable-Driven Parallel Robots, Springer, Cham, 2021, pp. 389–400.
[7] A.A. Kumar, J.-F. Antoine, G. Abba, Control of an underactuated 4 cable-driven parallel robot using modified input-output feedback linearization, IFAC-PapersOnLine 53 (2) (2020) 8777–8782.
[8] L. Scalera, A. Gasparetto, D. Zanotto, Design and experimental validation of a 3-DOF underactuated pendulum-like robot, IEEE/ASME Trans. Mechatronics 25 (1) (2020) 217–228.
[9] A. Berti, J.-P. Merlet, M. Carricato, Solving the direct geometrico-static problem of underconstrained cable-driven parallel robots by interval analysis, Int. J. Robot. Res. 35 (6) (2015) 723–739.
[10] M. Carricato, J.-P. Merlet, Stability analysis of underconstrained cable-driven parallel robots, IEEE Trans. Robot. 29 (1) (2013) 288–296.
[11] D. Surdilovic, J. Radojicic, Practical stability of under-constrained cable-suspended parallel robots, in: T. Bruckmann, A. Pott (Eds.), Cable-Driven Parallel Robots, Springer, Cham, 2019, pp. 85–98.
[12] E. Idà, T. Bruckmann, M. Carricato, Rest-to-rest trajectory planning for underactuated cable-driven parallel robots, IEEE Trans. Robot. 35 (6) (2019) 1338–1351.
[13] A. Lucarini, E. Idà, Kinematic modeling and design of a sensorized cable-routing system for cable-driven parallel robots, in: V. Niola, A. Gasparetto, G. Quaglia, G. Carbone (Eds.), Advances in Italian Mechanism Science, Springer, Cham, 2022, pp. 77–85.
[14] S. Gabaldo, E. Idà, M. Carricato, Pose-estimation methods for planar underactuated cable-driven parallel robots, in: S. Caro, A. Pott, T. Bruckmann (Eds.), Cable-Driven Parallel Robots, Springer, Cham, 2023, pp. 3–15.
[15] M. Zarei, A. Aflakian, A. Kalhor, 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, Mech. Mach. Theory 126 (2018) 377–396.
[16] M.R.J. Harandi, A. Molaei, H.D. Taghirad, Regulation control of a suspended cable-driven robot via energy shaping, in: 2021 60th IEEE Conference on Decision and Control, CDC, 2021, pp. 1438–1443.
[17] M.R.J. Harandi, S.A. Khalilpour, H. Taghirad, Robust IDA-PBC for a spatial underactuated cable driven robot with bounded inputs, in: 2021 29th Iranian Conference on Electrical Engineering, ICEE, 2021, pp. 689–694.
[18] M.R.J. Harandi, H.D. Taghirad, Adaptive interconnection and damping assignment passivity-based control for an underactuated cable-driven robot, Internat. J. Adapt. Control Signal Process. 35 (12) (2021) 2487–2498.
[19] J.-P. Merlet, Parallel Robots, Springer, 2012.
[20] V. Mattioni, E. Idà, M. Carricato, Force-distribution sensitivity to cable-tension errors in overconstrained cable-driven parallel robots, Mech. Mach. Theory 175 (104940) (2022) 1–14.
[21] E. Idà, D. Marian, M. Carricato, A deployable cable-driven parallel robot with large rotational capabilities for laser-scanning applications, IEEE Robot. Autom. Lett. 5 (3) (2020) 4140–4147.
[22] Y. Peng, W. Bu, Analysis of reachable workspace for spatial three-cable under-constrained suspended cable-driven parallel robots, J. Mech. Robot. 13 (6) (2021) 061002.
[23] J.-P. Merlet, Analysis of the influence of wires interference on the workspace of wire robots, in: J. Lenarčič, C. Galletti (Eds.), On Advances in Robot Kinematics, Springer, Dordrecht, 2004, pp. 211–218.
[24] S. Perreault, P. Cardou, C.M. Gosselin, M.J.-D. Otis, Geometric determination of the interference-free constant-orientation workspace of parallel cable-driven mechanisms, J. Mech. Robot. 2 (3) (2010).
[25] A. Martin, S. Caro, P. Cardou, Geometric determination of the cable-cylinder interference regions in the workspace of a cable-driven parallel robot, in: C. Gosselin, P. Cardou, T. Bruckmann, A. Pott (Eds.), Cable-Driven Parallel Robots, Springer, 2018, pp. 117–127.
[26] C.B. Pham, S.H. Yeo, G. Yang, I.-M. Chen, Workspace analysis of fully restrained cable-driven manipulators, Robot. Auton. Syst. 57 (9) (2009) 901–912.

[27] M. Gouttefarde, C.M. Gosselin, Analysis of the wrench-closure workspace of planar parallel cable-driven mechanisms, IEEE Trans. Robot. 22 (3) (2006) 434–445.

[28] M. Gouttefarde, D. Daney, J.-P. Merlet, Interval-analysis-based determination of the wrench-feasible workspace of parallel cable-driven robots, IEEE Trans. Robot. 27 (1) (2011) 1–13.

[29] J. Pusey, A. Fattah, S. Agrawal, E. Messina, Design and workspace analysis of a 6–6 cable-suspended parallel robot, Mech. Mach. Theory 39 (7) (2004) 761–778.

[30] G. Barrette, C.M. Gosselin, Determination of the dynamic workspace of cable-driven planar parallel mechanisms, J. Mech. Des. 127 (2) (2005) 242–248.

[31] D. Lin, G. Mottola, M. Carricato, X. Jiang, Modeling and control of a cable-suspended sling-like parallel robot for throwing operations, Appl. Sci. 10 (24) (2020).

[32] A.A. Kumar, J.-F. Antoine, P. Zattarin, G. Abba, Workspace analysis of a 4 cable-driven spatial parallel robot, in: V. Arakelian, P. Wenger (Eds.), ROMANSY 22 – Robot Design, Dynamics and Control, Springer, 2019, pp. 204–212.

[33] E. Idà, M. Carricato, A new performance index for underactuated cable-driven parallel robots, in: M. Gouttefarde, T. Bruckmann, A. Pott (Eds.), Cable-Driven Parallel Robots, Springer, Cham, 2021, pp. 24–36.

[34] E. Idá, M. Carricato, Static workspace computation for four-cable underactuated cable-driven parallel robots, in: IFToMM World Congress on Mechanism and Machine Science, Cham: Springer Nature Switzerland, 2023, pp. 588–598.

[35] R. Verhoeven, M. Hiller, S. Tadokoro, Workspace, stiffness, singularities and classification of tendon-driven stewart platforms, in: J. Lenarčič, M.L. Husty (Eds.), Advances in Robot Kinematics: Analysis and Control, Springer, 1998, pp. 105–114.

[36] M. Carricato, G. Abbasnejad, D. Walter, Inverse geometrico-static analysis of under-constrained cable-driven parallel robots with four cables, in: J. Lenarčič, M.L. Husty (Eds.), Latest Advances in Robot Kinematics, Springer, Dordrecht, 2012, pp. 365–372.

[37] E. Idà, S. Briot, M. Carricato, Robust trajectory planning of under-actuated cable-driven parallel robot with 3 cables, in: J. Lenarčič, B. Siciliano (Eds.), Advances in Robot Kinematics, Springer, Cham, 2021, pp. 65–72.

[38] A. Ben-Israel, T.N. Greville, Generalized Inverses: Theory and Applications, Springer, 2003.

[39] M. Carricato, Inverse geometrico-static problem of underconstrained cable-driven parallel robots with three cables, J. Mech. Robot. 5 (3) (2013) 11.

[40] A. Pott, Cable-Driven Parallel Robots: Theory and Application, Springer, 2018.

[41] J. Lamaury, M. Gouttefarde, A tension distribution method with improved computational efficiency, in: T. Bruckmann, A. Pott (Eds.), Cable-Driven Parallel Robots, Springer, Berlin, Heidelberg, 2013, pp. 71–85.

[42] F. Zaccaria, E. Idá, S. Briot, M. Carricato, Workspace computation of planar continuum parallel robots, IEEE Robot. Autom. Lett. 7 (2) (2022) 2700–2707.

[43] J. Nocedal, S.J. Wright, Numerical Optimization, Springer, 1999.

[44] E. Idà, S. Briot, M. Carricato, Identification of the inertial parameters of underactuated cable-driven parallel robots, Mech. Mach. Theory 167 (104504) (2022) 1–14.

[45] M. Gouttefarde, J. Lamaury, C. Reichert, T. Bruckmann, A versatile tension distribution algorithm for n -DOF parallel robots driven by $n+2$ cables, IEEE Trans. Robot. 31 (6) (2015) 1444–1457.

[46] R. Boumann, T. Bruckmann, Real-time cable force calculation beyond the wrench-feasible workspace, Robotics 9 (2) (2020).

[47] M. Angelini, E.I. Ida', D. Bertin, M. Carricato, E. Mantovani, D. Bazzi, V. Orassi, An underactuated cable-driven parallel robot for marine automated launch and recovery operations, in: International Design Engineering Technical Conferences and Computers and Information in Engineering Conference (Vol. 87363, p. V008T08A056), American Society of Mechanical Engineers, 2023.