Moving-Platform Pose Estimation for Cable-Driven Parallel Robots

Zane Zake[1,2], François Chaumette[3], Nicolò Pedemonte[2], and Stéphane Caro[1,4]

***Abstract*— Cable-Driven Parallel Robots (CDPRs) are parallel robots with rigid links replaced by cables. As for most parallel robots the determination of the analytical solutions to the direct geometrico-static model (DGSM) is a difficult task that is often not feasible online. However, the knowledge of the moving-platform (MP) pose is necessary in order to control the CDPR, e.g. with visual servoing. When the MP pose measurement is not available, an estimation can be sufficient. This paper compares three estimation methods: (a) control-based; (b) image-based; and (c) model-based. The three methods are implemented experimentally with an open-loop velocity controller and a closed-loop visual servoing controller. Overall, very good results are shown with model-based and control-based methods for both controllers. Finally, it is shown that the visual servoing controller leads to a better accuracy of the robot than the velocity controller.**

I. Introduction

Parallel robots have several kinematic chains composed of rigid links and lower-kinematic pairs connecting their end-effector, also known as the moving-platform (MP), to the base. In cable-driven parallel robots (CDPRs) the rigid links are replaced by cables. The MP motion is achieved by changing the cable lengths and that, in turn, is accomplished by winding and unwinding each cable on a winch, which is usually actuated by a rigidly fixed motor. CDPRs are characterized by a large translational workspace, a large payload capacity and a low mass in motion. These advantages lead to applications, such as: (a) moving large objects over large distances [1]; (b) moving objects with high velocity [2]; (c) providing feedback for a virtual reality application [3]; (d) assisting human rescue operations [4]. However, CDPRs are often lacking accuracy. In order to improve it one can enrich the mathematical models describing the geometric, kinematic, elasto-static and elasto-dynamic behavior of such robots [3]. In this case, CDPRs are controlled in their joint space. It is also possible to use exteroceptive sensors such as cameras to control CDPRs in their Cartesian space [5]–[9]. To improve the accuracy of a CDPR with respect to (wrt.) an object, the camera is usually mounted on the MP [5, 8, 9]. In this configuration the camera and the MP move simultaneously. When the camera approaches the object, it improves the accuracy that becomes excellent at the desired state if the object occupies a large part of the image plane. However, with the camera onboard the MP, the MP pose cannot be observed and thus it must be estimated.

In fact, the problem of knowing the MP pose is not exclusive to visual servoing with an onboard camera. Indeed, the solution(s) to the direct geometrico-static model (DGSM), whose aim is to determine the MP pose(s) and the corresponding tensions given a set of cable lengths, is a challenging task. For most parallel robots many solutions can exist. However, for CDPRs the complexity is even higher due to one-directional force that the cable can exert on the MP. The complexity of the problem has been highlighted by Carricato *et al.* for suspended CDPRs with less than six cables in [10]. More precisely, for a 3-cable CDPR the solution of DGSM is equivalent to finding the roots of a univariate polynomial of degree 156 [11] and of degree 216 for a 4-cable CDPR [12]. Finally, in 2015 the analysis was extended to a generic underconstrained CDPR with n cables [13].

The use of interval analysis in the DGSM has proven to be one of the most efficient methods [14, 15]. The proposed algorithm can be applied to both fully-constrained and suspended CDPRs and it provides the complete solution of the DGSM with all the MP poses that could be reached by the given cable length set. Furthermore, interval analysis can also be used for more complex CDPR models, for example with the sagging cable model [16]. However, it should be noted that finding all the solutions can have a high computational cost. Once all solutions are available, it is then necessary to determine the most probable one. Collard and Cardou propose to find the MP pose with the lowest equilibrium [17]. Indeed, for suspended CDPRs the most likely MP pose is the one where the MP center of mass is closest to the ground, which corresponds to the lowest equilibrium pose. The efficiency of the algorithm is proportional to the amount of cables of the suspended CDPR. A common work-around is to start at a known MP pose and use an iterative scheme to track that solution [18].

It is of interest to evaluate MP pose estimation methods that can be used online. For example, in visual servoing a new control output is computed every time an image becomes available from the camera, which is in general 30 to 60 times per second. Three different approaches are compared in this paper: (a) control-based; (b) image-based; (c) model-based. In the control-based approach the control output is integrated in order to estimate the new MP pose. This approach has been successfully used with different visual servoing controllers on CDPRs [8, 9]. In the image-based approach two images are used to measure the object

This work is supported by IRT Jules Verne (French Institute in Research and Technology in Advanced Manufacturing Technologies for Composite, Metallic and Hybrid Structures) in the framework of the Perform program.

[1]Laboratoire des Sciences du Numérique de Nantes, UMR CNRS 6004, 1, rue de la Noë, 44321 Nantes, France, Zane.Zake@ls2n.fr

[2]IRT Jules Verne, Chemin du Chaffault, 44340, Bouguenais, France, nicolo.pedemonte@irt-jules-verne.fr

[3]Inria, Univ Rennes, CNRS, IRISA, Rennes, France, Francois.Chaumette@inria.fr

[4]Centre National de la Recherche Scientifique (CNRS), 1, rue de la Noë, 44321 Nantes, France, stephane.caro@ls2n.fr

pose in the camera frame at two time instants. If at one time instant the MP pose is known, then at another it can be computed by using frame transformations, given that the object of interest does not move between iterations. Finally, in the model-based approach the lengths of the six cables that are most in tension are used to compute the MP pose in the vicinity of the previously known pose with the least squares method [19]. With all the approaches the initial MP pose needs to be known and the more accurate its knowledge, the better the overall behavior of the system. The three methods are implemented experimentally on a small CDPR with an open-loop velocity controller and a closed-loop visual servoing controller. Overall, we show that control-based and model-based methods produce good MP pose estimation on our CDPR. Model-based methods are slightly more accurate when used with an open-loop control, while the control-based method is the most accurate with visual servoing. Finally, it is shown that the visual servoing controller leads to a significantly better accuracy of the robot than the velocity controller.

This paper is organized as follows. Notations used throughout this paper are shown in Table I. Section II recalls the vision-based control strategy for a CDPR. Section III is dedicated to the introduction of different MP pose estimation methods. The experimental validation is shown in Section IV. Conclusions are drawn in Section V.

TABLE I

NOTATION USED THROUGHOUT THE PAPER

- $i = 1, \ldots, m$ denotes the ith cable, where $m = 8$.
- Boldface lowercase characters are vectors; boldface uppercase characters are matrices.
- $\mathcal{F}_b$, $\mathcal{F}_p$, $\mathcal{F}_c$, $\mathcal{F}_o$ denote the base, MP, camera and object frames respectively (resp.).
- ${}^i\mathbf{T}_j = \begin{bmatrix} {}^i\mathbf{R}_j & {}^i\mathbf{t}_j \\ \mathbf{0} & 1 \end{bmatrix}$ is the homogeneous transformation matrix from $\mathcal{F}_i$ to $\mathcal{F}_j$.
- $\widehat{\mathbf{A}}$ and $\hat{\mathbf{e}}$ are the estimations of $\mathbf{A}$ and $\mathbf{e}$, resp.
- $[\mathbf{e}]_\times$ denotes the cross-product matrix of vector $\mathbf{e}$.
- ${}^i\mathbf{a}$ is the vector $\mathbf{a}$ expressed in $\mathcal{F}_i$
- A_i and B_i are the exit and the anchor points of the ith cable, resp.
- ${}^b\mathbf{a}_i$ and ${}^p\mathbf{b}_i$ are the vectors pointing from origin of $\mathcal{F}_b$ and $\mathcal{F}_p$ to A_i and B_i, resp.

II. VISUAL SERVOING OF A CDPR

The schematic of a spatial CDPR with an onboard camera and a target object in the workspace can be seen in Fig. 1. In our case, the six degrees of freedom (DoF) of the MP are constrained with eight cables. The onboard camera is fixed so that it can observe the object in the workspace. The corresponding visual servoing (VS) control scheme is shown in Fig. 2.

At every iteration an image is received from the onboard camera and is processed by a computer vision algorithm, which then outputs the current value of a feature vector $\mathbf{s}$. The contents of $\mathbf{s}$ depend on the chosen VS approach.

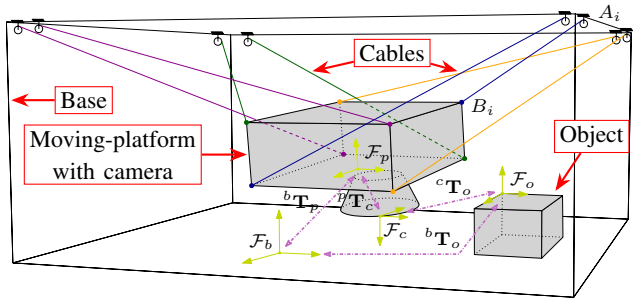


Fig. 1. Schematic of a spatial CDPR with the relevant frames

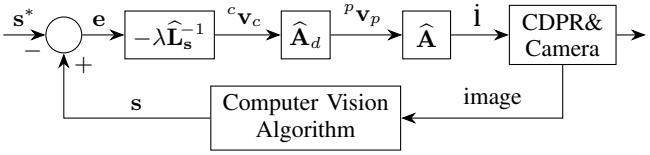


Fig. 2. Control scheme for visual servoing of a CDPR

For example, it can simply be the target pose expressed in the camera frame $\mathcal{F}_c$ when a Pose-Based Visual Servoing (PBVS) is implemented [20].

An error $\mathbf{e}$ is computed by comparing the feature vector $\mathbf{s}$ and its desired value $\mathbf{s}^*$. To decrease the error $\mathbf{e}$, an exponential decoupled form is selected

$$\dot{\mathbf{e}} = -\lambda \mathbf{e} \tag{1}$$

where λ is a positive gain. The camera velocity is then expressed as:

$${}^c\mathbf{v}_c = -\lambda \widehat{\mathbf{L}}_\mathbf{s}^\dagger \mathbf{e} \tag{2}$$

where $\widehat{\mathbf{L}}_\mathbf{s}^\dagger$ is the Moore-Penrose pseudo-inverse of the estimation of the interaction matrix $\mathbf{L}_\mathbf{s}$, defined such that $\dot{\mathbf{s}} = \mathbf{L}_\mathbf{s} {}^c\mathbf{v}_c$, and thus it depends on the chosen VS approach.

The camera velocity is then transformed into the MP twist:

$${}^p\mathbf{v}_p = \widehat{\mathbf{A}}_d \, {}^c\mathbf{v}_c \tag{3}$$

where $\widehat{\mathbf{A}}_d$ is the estimation of the adjoint matrix $\mathbf{A}_d$ [21]:

$$\mathbf{A}_d = \begin{bmatrix} {}^p\mathbf{R}_c & [{}^p\mathbf{t}_c]_\times {}^p\mathbf{R}_c \\ \mathbf{0}_3 & {}^p\mathbf{R}_c \end{bmatrix} \tag{4}$$

The cable velocity vector $\dot{\mathbf{l}}$ is related to the MP twist ${}^p\mathbf{v}_p$:

$$\dot{\mathbf{l}} = \widehat{\mathbf{A}} \, {}^p\mathbf{v}_p \tag{5}$$

where $\mathbf{A}$ is the Forward Jacobian matrix and it takes the following form when expressed in the MP frame $\mathcal{F}_p$ [19]:

$$\mathbf{A} = \begin{bmatrix} {}^p\mathbf{u}_1^\top & ({}^p\mathbf{b}_1 \times {}^p\mathbf{u}_1)^T \\ \vdots & \vdots \\ {}^p\mathbf{u}_m^\top & ({}^p\mathbf{b}_m \times {}^p\mathbf{u}_m)^T \end{bmatrix} \tag{6}$$

where ${}^p\mathbf{u}_i$ is the unit vector of $\mathbf{l}_i$, pointing from the exit point A_i to the anchor point B_i of the ith cable, and it is expressed as follows:

$${}^p\mathbf{u}_i = \frac{\mathbf{l}_i}{\|\mathbf{l}_i\|_2} = \frac{{}^p\mathbf{b}_i - {}^p\mathbf{a}_i}{\|\mathbf{l}_i\|_2} = \frac{{}^p\mathbf{b}_i - {}^p\mathbf{R}_b \, {}^b\mathbf{a}_i - {}^p\mathbf{t}_b}{\|\mathbf{l}_i\|_2} \tag{7}$$

As can be seen, the MP pose is required to compute ${}^{p}\mathbf{u}_i$.

The output of the control scheme, that is, the cable velocity vector $\dot{\mathbf{l}}$, is expressed by injecting (2) and (3) into (5):

$$\dot{\mathbf{l}} = -\lambda\, \widehat{\mathbf{A}}\, \widehat{\mathbf{A}}_d\, \widehat{\mathbf{L}}_{\mathbf{s}}^{\dagger}\, \mathbf{e} \tag{8}$$

Thus, the computation of the control output is not possible without a measurement or an estimation of the MP pose.

III. Moving-platform pose estimation

As recalled in the introduction and described in [8, 9], it is indeed possible to coarsely estimate the MP pose by successive integration of the control output. Here, we propose to compare three different MP pose estimation methods, namely: (i) control-based; (ii) image-based; (iii) model-based. In all three cases it is assumed that the initial MP pose ${}^{b}\mathbf{p}_{p_0}$ at $t = 0\,\mathrm{s}$ is known.

A. Control-Based Estimation

Assuming that the CDPR is controlled in velocity and that it is able to produce the velocity prescribed by the controller, this control output can be used for the MP pose estimation. More precisely, at every iteration the homogeneous transformation matrix ${}^{b}\mathbf{T}_p$ can be updated as follows:

$${}^{b}\mathbf{T}_p(t) = {}^{b}\mathbf{T}_p(t - \Delta t)\, \mathrm{e}^{({}^{p}\mathbf{v}_p, \Delta t)} \tag{9}$$

where $\mathrm{e}^{({}^{p}\mathbf{v}_p, \Delta t)}$ is the exponential map [22] given a velocity ${}^{p}\mathbf{v}_p$ and a time interval Δt. Note that the MP velocity ${}^{p}\mathbf{v}_p$ is one of the control outputs, as can be seen from Fig. 2.

B. Image-Based Estimation

Assuming that the transformation ${}^{c}\mathbf{T}_o$ between the camera and object frames can be estimated, like in PBVS, it is possible to express the object pose in the base frame $\mathcal{F}_b$ as:

$${}^{b}\mathbf{T}_o = {}^{b}\mathbf{T}_p\, {}^{p}\mathbf{T}_c\, {}^{c}\mathbf{T}_o \tag{10}$$

The transformation matrices present in (10) and the corresponding frames are shown in Fig. 1. Since the camera is fixed on the MP, then ${}^{p}\mathbf{T}_c$ does not change with time.

At two different iterations (10) can be expressed as:

$$\begin{cases} {}^{b}\mathbf{T}_o(t_0) = {}^{b}\mathbf{T}_p(t_0)\, {}^{p}\mathbf{T}_c\, {}^{c}\mathbf{T}_o(t_0) \\ {}^{b}\mathbf{T}_o(t) = {}^{b}\mathbf{T}_p(t)\, {}^{p}\mathbf{T}_c\, {}^{c}\mathbf{T}_o(t) \end{cases} \tag{11}$$

If the object is motionless, then ${}^{b}\mathbf{T}_o$ does not change with time, thus ${}^{b}\mathbf{T}_o(t_0) = {}^{b}\mathbf{T}_o(t)$ and then ${}^{b}\mathbf{T}_p(t)$ can be expressed from (11) as:

$${}^{b}\mathbf{T}_p(t) = {}^{b}\mathbf{T}_p(t_0)\, {}^{p}\mathbf{T}_c\, {}^{c}\mathbf{T}_o(t_0)\, {}^{c}\mathbf{T}_o^{-1}(t)\, {}^{p}\mathbf{T}_c^{-1} \tag{12}$$

Unlike control integration, this approach does not require the CDPR to achieve the control velocity. Indeed, the calculation is done without taking into account the velocity and time spent to achieve the current pose. Instead two separate object pose measurements are used. For that, we can use two successive measurements so that $t_0 = t - \Delta t$. In fact, the current measurement can be compared to any previous measurement, for example the very first one. For the results presented in Section IV, we use two successive measurements.

C. Model-Based Estimation

The Direct Geometric Model (DGM) expresses the MP pose as a function of cable lengths. For a spatial CDPR with 8 cables this is a very complex problem, which often cannot be solved rapidly enough to be used in control. As already said in the introduction, special approaches such as interval analysis have been used in [14]–[16] to obtain all the solutions to the DGSM, however this usually comes at a high computational cost. Furthermore, it is then necessary to determine which is the correct solution.

Using cable tension measurements, it is possible to estimate the MP pose given the previous pose and cable lengths. Several approaches are possible depending on the selected CDPR model complexity.

1) Simple CDPR model: In the simplest CDPR model the cables are assumed to be massless, non-elastic and always straight. In such a case, at any given time at most six cables can be in tension [23]. Thus, the DGM can be simplified to only take into account the six cables that are most in tension. Such a model is similar to the Gough-Stewart platform, whose DGM can have up to 40 solutions. The computation of all of the solutions can be a lengthy process, which is not necessary in this case. Indeed, we are only interested in the solution that is in the vicinity of the last known pose. In this case, it is possible to use the least squares optimization method to find the MP pose [19]. The estimation procedure is shown in Algorithm 1.

Algorithm 1: MP pose estimation from cable lengths

1: **Initialization**
2: Define the CDPR model with the Cartesian coordinates of A_i and B_i expressed in $\mathcal{F}_b$ and $\mathcal{F}_p$, resp.
3: Define the initial MP pose and record it in ${}^{b}\mathbf{p}_{p_o}$
4: **End of Initialization**
5: **Pose Estimation**
6: **while** True **do**
7: Get measurements: cable tensions $\boldsymbol{\tau}$ and lengths $\mathbf{l}$
8: Find the six largest cable tensions
9: Compose the residual equations for these six cables: $f_i = ||{}^{b}\mathbf{b}_i - {}^{b}\mathbf{a}_i||_2 - l_i$
10: Compose the equation to normalize the quaternions: $f_7 = q_1^2 + q_2^2 + q_3^2 + q_4^2 - 1$
11: Find ${}^{b}\mathbf{p}_p$ with the least squares method, giving ${}^{b}\mathbf{p}_{p_o}$ as the initial guess and f_1 to f_7 as the system to be minimized
12: Update ${}^{b}\mathbf{p}_{p_o} = {}^{b}\mathbf{p}_p$ for next iteration
13: **end while**
14: **End of Pose Estimation**

Note that the representation of the MP pose is important, because it affects the amount of unknowns. Here, the pose is expressed as ${}^{b}\mathbf{p}_p = \begin{bmatrix} x, & y, & z, & q_1, & q_2, & q_3, & q_4 \end{bmatrix}$, where $\begin{bmatrix} x, & y, & z \end{bmatrix}$ is the translational part and $\begin{bmatrix} q_1, & q_2, & q_3, & q_4 \end{bmatrix}$ is the rotational part expressed as quaternions. Note that, when using quaternions, their normalizing equation $q_1^2 + q_2^2 + q_3^2 + q_4^2 = 1$ must also be taken into account. Thus, there are 7 unknowns and

7 equations, which makes the system solvable.

The least squares algorithm to compute the MP pose is fast (for example the implementation in Python using *scipy.least_squares* takes about 0.009 s). Furthermore, using only the length of the cables that are most in tension makes the algorithm robust to cable slackness, because the slack cables are ignored. However, it is a model-based estimation method, meaning that any error in the model will lead to MP pose estimation errors.

2) Elastic cables: Cable elasticity affects the final MP pose as well as the tensions that are obtained in the cables. Indeed, only with elastic cables it is possible to have all eight of them in tension [23].

When considering elasticity, the ith cable tension at time t can be expressed as:

$$\tau_i(t) = k(t)\,\delta l_i + \tau_i(0) \tag{13}$$

where:

- δl_i is the elongation of the ith cable;
- $\tau_i(0)$ is the ith cable tension with the MP positioned on the ground;
- $k(t) = ES/l_i(t)$ is the cable stiffness;
- E is the Young modulus of the ith cable;
- S is the cross-section of the ith cable;
- $l_i(t)$ is the length of the ith cable at time t.

Assuming that the tension measurements are available, cable elongation δl_i can be expressed from (13) as:

$$\delta l_i = \frac{\big(\tau_i(t) - \tau_i(0)\big) l_i(t)}{ES} \tag{14}$$

Thus, the ith cable length l_i obtained from the ith motor encoder can be corrected to $\widehat{l_i} = l_i + \delta l_i$. The estimation of the MP pose is done as described in Algorithm 1, but in the equation on line 9 the cable length l_i is substituted by $\widehat{l_i}$.

3) Pulley kinematics: For pulleys of non-negligible size, the cable exit point has to be defined as the tangent point between the cable and the pulley sheave. The geometric model of a pulley is shown in Fig. 3. Here, the frame $\mathcal{F}_i$ of the ith pulley has its origin in point A_i and the axes are $\mathbf{x}_i$, $\mathbf{y}_i$ and $\mathbf{z}_i$. Axis $\mathbf{z}_i$ is vertical, $\mathbf{x}_i$ goes through A_i and the center of the pulley P_i, and $\mathbf{y}_i = \mathbf{z}_i \times \mathbf{x}_i$. The ith cable and the corresponding pulley sheave lie in the plane spanned by vectors $\mathbf{x}_i$ and $\mathbf{z}_i$.

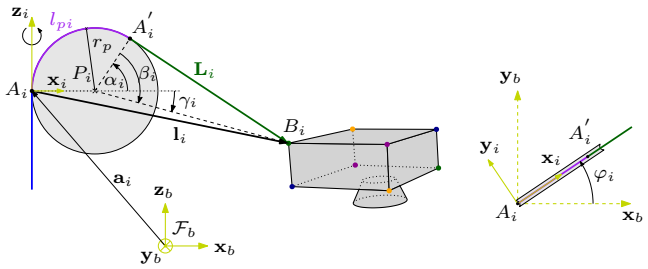


Fig. 3. Pulley geometry

The actual ith cable exit point is the tangent point to its pulley A_i'. The vector pointing from P_i to A_i' is named $\mathbf{c}_i$ while the vector pointing from P_i to B_i is named $\mathbf{m}_i$. The exit point location on the pulley depends on the MP pose. L_i is the length of the ith cable between points A_i' and B_i. l_{pi} denotes the length of the cable wrapped on the pulley between points A_i and A_i'. The full cable length from A_i to B_i that will be used in Algorithm 1 on line 9 becomes [24]:

$$l_{fi} = L_i + l_{pi} \tag{15}$$

with

$$l_{pi} = r_p\big(\pi - \alpha_i\big) \tag{16}$$

and the angle α_i is computed as:

$$\alpha_i = -\beta_i + \gamma_i \tag{17}$$

where:

$$\beta_i = -\text{atan2}(L_i,\, r_p) \tag{18}$$

and

$$\gamma_i = \arcsin\left(\frac{{}^b a_{iz} - {}^b b_{iz}}{||\mathbf{m}_i||_2}\right) \tag{19}$$

Note that here ${}^b a_{iz}$ and ${}^b b_{iz}$ are the third component of vectors ${}^b\mathbf{a}_i$ and ${}^b\mathbf{b}_i$, respectively.

The angle φ_i denotes the rotation of pulley plane about the axis $\mathbf{z}_i$. It can be computed as:

$$\varphi_i = \text{atan2}(l_{yi}, l_{xi}) \tag{20}$$

where l_{xi} and l_{yi} are the components of $\mathbf{l}_i$ along the $\mathbf{x}$ and $\mathbf{y}$ axes of the base frame $\mathcal{F}_b$. The vector ${}^b\mathbf{n}_i$ points from the origin of the base frame $\mathcal{F}_b$ to the pulley center-point P_i and is computed as:

$${}^b\mathbf{n}_i = {}^b\mathbf{a}_i + r_p\,{}^b\mathbf{R}_i\,\mathbf{x}_b = {}^b\mathbf{a}_i + r_p\,{}^b\mathbf{x}_i \tag{21}$$

with

$${}^b\mathbf{R}_i = \mathbf{R}_{zi}(\varphi_i) = \begin{bmatrix} \cos(\varphi_i) & -\sin(\varphi_i) & 0 \\ \sin(\varphi_i) & \cos(\varphi_i) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{22}$$

and r_p is the radius of the pulley sheave.

Vector $\mathbf{m}_i$ is computed as:

$$\mathbf{m}_i = {}^b\mathbf{b}_i - {}^b\mathbf{n}_i \tag{23}$$

The ith cable length L_i can be computed by:

$$L_i = \sqrt{\mathbf{m}_i\,\mathbf{m}_i^\top - r_p^2} \tag{24}$$

4) Elastic cables and pulley kinematics: Finally, we can also combine pulley kinematics and cable elasticity in the MP pose estimation. The full cable length from A_i to B_i that will be used in Algorithm 1 on line 9 becomes:

$$\hat{l}_{fi} = \hat{L}_i + l_{pi} = L_i + \delta L_i + l_{pi} \tag{25}$$

where L_i, l_{pi} and δL_i are computed by (24), (16), and (14) resp.

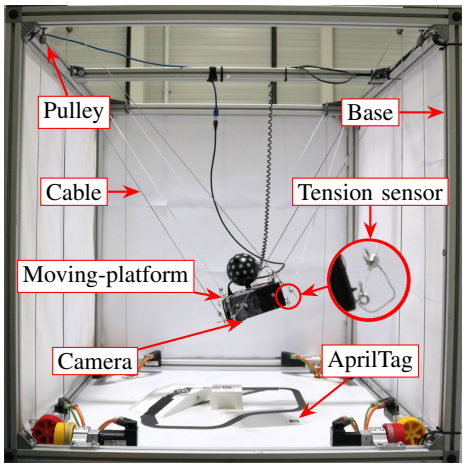


Fig. 4. ACROBOT: a CDPR prototype located at IRT Jules Verne, Nantes

IV. Experimental Validation

A. Experimental setup

The CDPR used for experimental validation is shown in Fig. 4. It is a six-DoF suspended CDPR with eight cables, named ACROBOT. Its frame is a $1.2\,\text{m} \times 1.2\,\text{m} \times 1.2\,\text{m}$ cube. The MP size is $0.18\,\text{m} \times 0.17\,\text{m} \times 0.07\,\text{m}$ and its mass is $3.5\,\text{kg}$. The cables are Dyneema SK78 $\varnothing 2\,\text{mm}$ with $E = 111\,\text{GPa}$ [25]. Tension sensors are mounted on the cables close to their anchor points B_i. They are calibrated for a range from $-25\,\text{N}$ to $75\,\text{N}$, with an accuracy of $0.24\,\text{N}$ and repeatability of $0.01\,\text{N}$.

Ground-truth measurements are done by a Creaform C-Track. It is a dynamic tracking system with accuracy of $0.1\,\text{mm}$ and repeatability of $0.02\,\text{mm}$. For the image-based estimation method we use an AprilTag [26] as an object. It is a fiducial marker that is recognized and localized by the ViSP library [27], thus a measurement of ${}^{c}\mathbf{T}_{o}$ is available at each iteration.

The estimation methods are defined as follows:

- Est. 1 is the control-based method;
- Est. 2 is the image-based method;
- Est. 3 is the model-based method with the simple model;
- Est. 4 is the model-based method with cable elasticity;
- Est. 5 is the model-based method with pulley geometry;
- Est. 6 is the model-based method with cable elasticity and pulley geometry.

B. Results with open-loop trajectory

A simple open-loop controller is implemented to generate a straight-line trajectory between the following points:

- ${}^{b}\mathbf{p}_{p_0} = [0.112\,\text{m},\ -0.032\,\text{m},\ 0.301\,\text{m},\ -12°,\ -10°,\ 0°]$
- ${}^{b}\mathbf{p}_{p_1} = [0.284\,\text{m},\ -0.197\,\text{m},\ 0.08\,\text{m},\ 0°,\ 0°,\ 0°]$
- ${}^{b}\mathbf{p}_{p_2} = [0.284\,\text{m},\ -0.197\,\text{m},\ 0.25\,\text{m},\ 0°,\ 0°,\ 0°]$

Initially the MP is at ${}^{b}\mathbf{p}_{p_0}$ and a trajectory is generated to reach ${}^{b}\mathbf{p}_{p_1}$; then height along z-axis is increased to reach ${}^{b}\mathbf{p}_{p_2}$; and finally the MP returns to ${}^{b}\mathbf{p}_{p_0}$.

The trajectory between two poses, named $\mathbf{p}_s$ and $\mathbf{p}_f$, is generated using a fifth-order polynomial [28]:

$$s = bt^5 + ct^4 + dt^3 + et^2 + ft + g \tag{26}$$

while considering the following conditions at the start and at the end of the trajectory: $s(t_s) = \dot{s}(t_s) = \ddot{s}(t_s) = 0$, $s(t_f) = 1$, and $\dot{s}(t_f) = \ddot{s}(t_f) = 0$.

The MP position ${}^{b}\mathbf{t}_p$ and translational velocity ${}^{b}\boldsymbol{v}_p$ as a function of time are expressed as:

$$\begin{cases} {}^{b}\mathbf{t}_p(t) = \mathbf{t}_s + (\mathbf{t}_f - \mathbf{t}_s)\, s(t) & \text{(27a)} \\ {}^{b}\boldsymbol{v}_p(t) = (\mathbf{t}_f - \mathbf{t}_s)\, \dot{s}(t) & \text{(27b)} \end{cases}$$

where $\mathbf{t}_s$ and $\mathbf{t}_f$ are the translational parts of $\mathbf{p}_s$ and $\mathbf{p}_f$, resp.

As for the rotation, we start from ${}^{p_s}\mathbf{R}_{p_f} = {}^{b}\mathbf{R}_{p_s}^{\top}\, {}^{b}\mathbf{R}_{p_f}$, where ${}^{b}\mathbf{R}_{p_s}$ and ${}^{b}\mathbf{R}_{p_f}$ are the rotation matrices for poses $\mathbf{p}_s$ and $\mathbf{p}_f$, resp. When changed to axis-angle representation, ${}^{p_s}\mathbf{R}_{p_f}$ is noted as $\theta_p \mathbf{u}_p$. Here, the unit vector $\mathbf{u}$ is constant and the angle θ_p is a function of time $\theta_p(t) = \theta_p s(t)$, which corresponds to a rotation matrix ${}^{p_s}\mathbf{R}_{p_{curr}}$ and allows us to compute the current rotation matrix of the MP:

$${}^{b}\mathbf{R}_{p_{curr}} = {}^{b}\mathbf{R}_{p_s}\, {}^{p_s}\mathbf{R}_{p_{curr}} \tag{28}$$

Finally, the angular velocity ${}^{b}\boldsymbol{\omega}_p$ is computed as:

$${}^{b}\boldsymbol{\omega}_p(t) = \theta_p \mathbf{u}_p \dot{s}(t) \tag{29}$$

Then the cable velocities are computed as:

$$\dot{\mathbf{l}} = \widehat{\mathbf{A}}\, {}^{b}\mathbf{v}_p(t) \tag{30}$$

where ${}^{b}\mathbf{v}_p(t) = \left[{}^{b}\boldsymbol{v}_p^{\top}(t) \quad {}^{b}\boldsymbol{\omega}_p^{\top}(t)\right]^{\top}$ is given from (27b) and (29). Note that in (30), the Jacobian matrix $\widehat{\mathbf{A}}$ uses the MP pose computed from (27a) and (28). The control frequency is $25\,\text{Hz}$.

The experimental results are shown in Figs. 5 and 6 (please also see the attached video). The task was executed once using controller (30), during which all data were recorded to be able to compare the different estimation methods afterwards. The planned MP trajectory is shown in cyan in Fig. 5a. The MP trajectory was measured by C-Track and it is shown in blue in Fig. 5a. Note that while the initial pose ${}^{b}\mathbf{p}_{p_0}$ is indeed where the experiment starts, however the poses ${}^{b}\mathbf{p}_{p_1}$ and ${}^{b}\mathbf{p}_{p_2}$ are not reached accurately. At the end of the first part of the trajectory the MP passes ${}^{b}\mathbf{p}_{p_1}$ and comes to a stop right afterwards. Then only the z component is changed, thus ${}^{b}\mathbf{p}_{p_2}$ is also not achieved accurately. Finally, the MP accurately returns to the initial pose ${}^{b}\mathbf{p}_{p_0}$. Indeed, as can be seen in Figs. 6a and 6d the initial and final images received from the camera are identical. Thus, it can be concluded that the CDPR executes the control velocity $\dot{\mathbf{l}}$ precisely. The positioning error in ${}^{b}\mathbf{p}_{p_1}$ and ${}^{b}\mathbf{p}_{p_2}$ can also be seen in Fig. 6. Ideally with the MP in ${}^{b}\mathbf{p}_{p_1}$ the AprilTag should be centered in the image and with no orientation about z axis. However, as can be seen in Fig. 6b the AprilTag is not centered and it is rotated. Similarly, in Fig. 6c the AprilTag is also rotated. These positioning errors come from the fact that the analytical form of the Jacobian matrix $\mathbf{A}$ used in (30) is

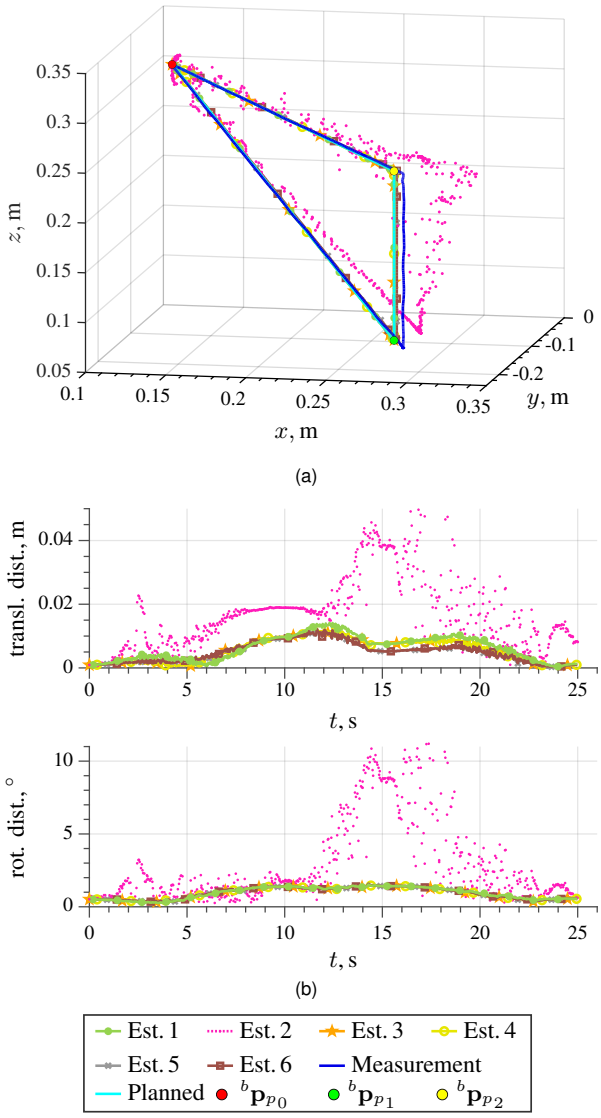


Fig. 5. Experimental results: (a) MP trajectory as estimated by the different estimation methods, as planned by the open-loop controller and as measured by C-Track; (b) deviation of the MP pose estimation from the measurement by C-Track

based on a too simplified CDPR model not corresponding to the real robot, due to calibration errors and approximations in the model, e.g., errors in the cable anchor and exit point coordinates, uncertainties in cable elongation, etc.

After executing the task, the recorded values were supplied to all MP pose estimation methods. The resulting trajectories are shown in Fig. 5 as Est. 1 to Est. 6. Surprisingly, the estimated trajectory with control-based and model-based estimators corresponds to the planned one (but not to the achieved one). For instance, at the end of the first part of the trajectory, the estimated MP pose corresponds to ${}^b\mathbf{p}_{p_1}$, even though the MP is actually shifted by approximately 1 cm. Similarly, at the end of the second part of the trajectory the estimated MP pose corresponds to ${}^b\mathbf{p}_{p_2}$. This means that all sampling effects are fully negligible in implementing (30), leading the robot to achieve the computed velocity ${}^b\mathbf{v}_p(t)$

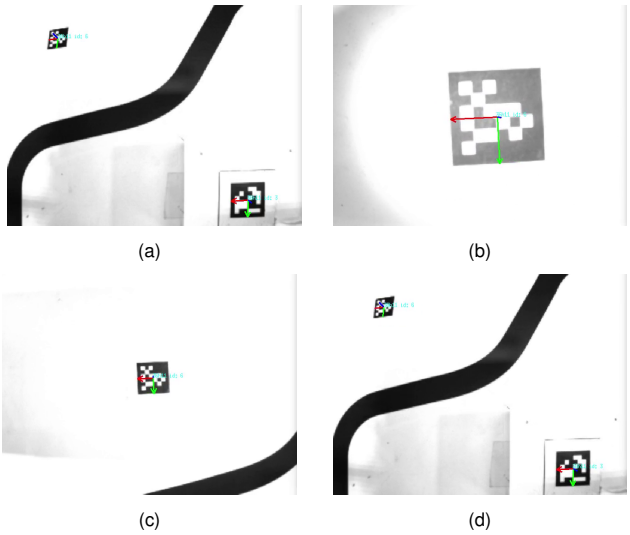


Fig. 6. Images from the onboard camera: (a) at $t = 0\,s$ at ${}^b\mathbf{p}_{p_0}$; (b) at $t = 10\,\mathrm{s}$ when ${}^b\mathbf{p}_{p_1}$ is supposed to be reached; (c) at $t = 15\,\mathrm{s}$ when ${}^b\mathbf{p}_{p_2}$ is supposed to be reached; (d) at $t = 25\,\mathrm{s}$ when ${}^b\mathbf{p}_{p_0}$ is reached again

and reaching the desired cable length values by successive integration of $\dot{\mathbf{l}}$. Finally, as the model-based methods rely on a model, it is not surprising its results deviate from the ground-truth. Similarly, as the control-based method is not based on any measurement, it is not surprising its results also deviate from the ground-truth.

Fig. 5b allows us to evaluate the accuracy of estimation with each method. Here, the deviation of the estimation from the actual MP pose measurement is shown as the translational and the rotational distance. Note that the rotational distance is the angle θ from the axis-angle expression of the rotation matrix between the measured and the estimated MP poses. The vertical dotted lines at $t = 10\,\mathrm{s}$ and $t = 15\,\mathrm{s}$ correspond to the end of the first and second parts of the trajectory, when ${}^b\mathbf{p}_{p_1}$ and ${}^b\mathbf{p}_{p_2}$ are supposed to be reached.

Let us begin with Est. 2, because it is the only one producing a significantly different result. Est. 2 corresponds to the image-based MP pose estimation method. It is clear that the AprilTag used in this experiment was too small to produce a good measurement. Indeed, note the good accuracy of the measurement between $t = 8\,\mathrm{s}$ and $t = 12\,\mathrm{s}$, when the MP is the closest to the AprilTag. On the other hand, as the AprilTag becomes significantly smaller in the image, the estimation of the MP pose becomes highly noisy. Thus, with the image-based method the accuracy of the MP pose estimation depends on the distance between the camera and the object and thus its size in the image.

The control-based estimation Est. 1 is shown in green. The translational deviation reaches $1.4\,\mathrm{cm}$ at $t = 12.4\,\mathrm{s}$ and the task is finished with a deviation of only $1.3\,\mathrm{mm}$. The estimation of the MP orientation does not surpass $1.6°$ error and the task is finished with a $0.55°$ error. The result with the model-based methods is almost the same. More precisely, the largest translational deviation is $1.21\,\mathrm{cm}$ at $t = 11.6\,\mathrm{s}$ and at the end of the task reduces to just $1\,\mathrm{mm}$, because the

MP returns to its initial pose ${}^b\mathbf{p}_{p_0}$. The largest rotational distance is $1.6°$ at $t = 14.3\,\text{s}$ and it reduces to $0.55°$ by the end of the task. Note that the results with all four model-based approaches are almost identical. Indeed, for a small CDPR with Dyneema cables that are almost inelastic and very small pulleys of 9 mm in diameter, the output of the four model-based estimation methods is almost the same.

C. Results with PBVS

The second part of the experiments consists of repeating the trajectory from ${}^b\mathbf{p}_{p_0}$ to ${}^b\mathbf{p}_{p_1}$ with a PBVS controller implemented at 25 Hz. The corresponding AprilTag poses in the camera frame are the following:

- ${}^c\mathbf{p}_{o0} = [-0.123\,\text{m}; -0.101\,\text{m}; 0.338\,\text{m}; 168°; -8°; -179°]$
- ${}^c\mathbf{p}_{o1} = [0\,\text{m}; 0\,\text{m}; 0.09\,\text{m}; 180°; 0°; 180°]$

and the feature vector $\mathbf{s}$ is defined as $\mathbf{s} = ({}^c\mathbf{t}_o, \theta\mathbf{u})$, where $\theta\mathbf{u}$ is the axis angle representation of the rotation matrix ${}^{c^*}\mathbf{R}_c$ between the desired and the current camera frames. In this case, we use each estimation method within the control scheme shown in Fig. 2, thus the experiment needs to be done separately for each of them. The results are shown in Fig. 7. Note that the trajectory of the MP in Fig. 7 is no longer a straight line due to the choice of our feature vector $\mathbf{s}$. Indeed, it ensures a straight line trajectory of the target object in the image, but does not control the camera, nor the MP trajectory.

In Fig. 7a trajectories generated when using control-based and model-based MP pose estimation appear identical. Indeed, only the pink curve, corresponding to image-based estimation Est. 2, is slightly deviated. As can be seen in Fig. 7b, the image-based estimation still provides the worst results with a rather noisy MP pose estimation. The quality of the model-based estimations is very similar to the previous experiment. The deviation does not surpass 1.29 cm and $2°$ for Est. 3 and Est. 4. The result is slightly worse for the model-based methods that take pulley kinematics into account (Est. 5 and Est. 6), where the translational deviation reaches 1.56 cm. Unlike with the open-loop controller, here there are differences between the control-based and the model-based estimation accuracy. Indeed, when used with a closed-loop controller the control-based method provides the best MP pose estimation accuracy, not surpassing 0.9 cm and $2°$. As we now use a closed-loop controller, the controller adapts its output to all perturbations, including the misestimation of the MP pose. For this reason, the control-based MP pose estimation gives better results. Thus, the choice of the MP pose estimation method depends on the choice of the controller.

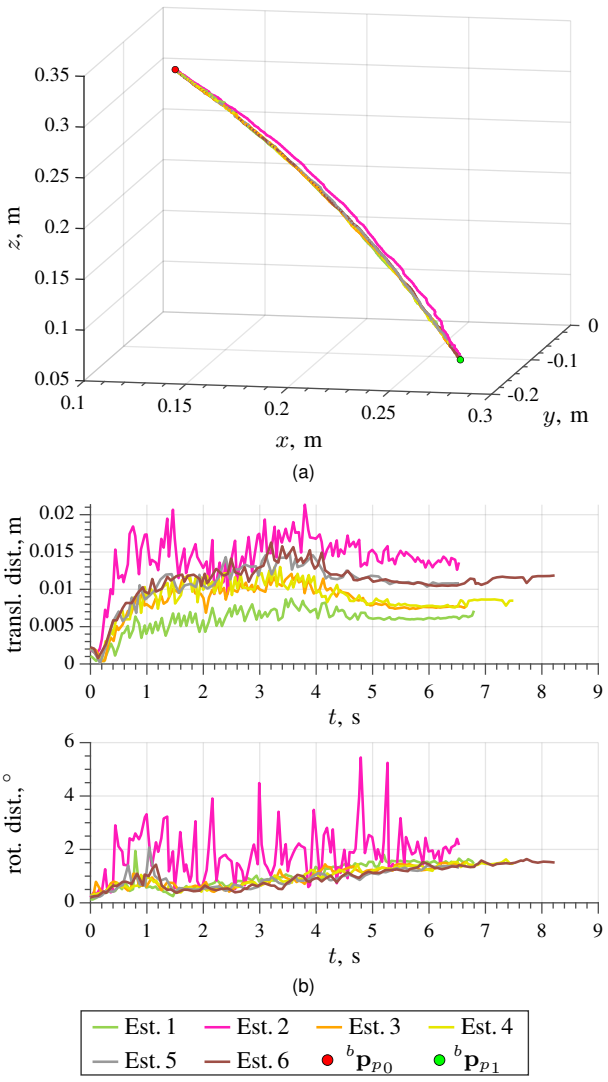


Fig. 7. Experimental results with PBVS: (a) MP trajectory measured by C-Track; (b) deviation of the MP pose estimation from the measurement by C-Track

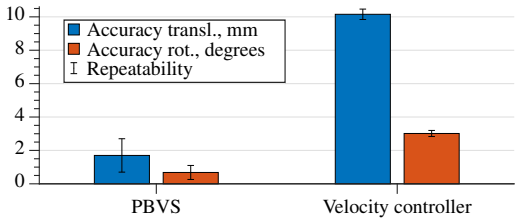


Fig. 8. MP pose accuracy at ${}^b\mathbf{p}_{p_1}$

D. Accuracy and Repeatability

Finally, as two different controllers were used to arrive at ${}^b\mathbf{p}_{p_1}$ from ${}^b\mathbf{p}_{p_0}$, it is possible to evaluate the accuracy achieved by these controllers. The bar-graph is shown in Fig. 8. To evaluate repeatability the experiment with the open-loop velocity controller was repeated 10 times. Similarly, the experiment with the PBVS controller was repeated 30 times. More precisely, five repetitions were done using each of the six MP pose estimation methods in the PBVS control loop. When using PBVS the translational accuracy is five times better, while the rotational accuracy is four times better than with the open-loop velocity controller. Indeed, as the AprilTag is perceived, the task is only finished when the current object pose converges to the desired one and thus a high accuracy can be achieved.

V. Conclusions

The knowledge of the MP pose is necessary to be able to control a CDPR. Three different MP pose estimation methods were compared in this paper. The image-based method shows the noisiest results with both controllers. Here, accuracy of MP pose estimation depends on the distance between the camera and the target object. The smaller the distance, the better the accuracy. Control-based and model-based methods show very good results with both controllers. Model-based methods are slightly superior when used with an open-loop controller. Note that for this CDPR there is no difference as to which model is used in the model-based method. On the other hand, the control-based method produced a better result for the closed-loop controller. It is also the simplest to implement. Thus, the choice of the MP pose estimation method depends on the choice of the controller.

Note that the CDPR used in the experimental validation is almost perfect, that is, its controller is able to achieve the computed velocities. Thus it would be of interest to extend this study by implementing the same controllers and estimation methods on larger CDPRs with a high payload that are not able to ensure this nice property.

Finally, the accuracy of the controllers was compared. The accuracy is four to five times better with PBVS than with the velocity controller.

References

[1] J. Albus, R. Bostelman, and N. Dagalakis, "The NIST SPIDER, a robot crane," *Journal of research of the National Institute of Standards and Technology*, vol. 97, no. 3, p. 373, 1992.

[2] 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.

[3] A. Fortin-Côté, P. Cardou, and C. Gosselin, "An admittance control scheme for haptic interfaces based on cable-driven parallel mechanisms," in *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 819–825, 2014.

[4] J. P. Merlet and D. Daney, "A portable, modular parallel wire crane for rescue operations," in *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 2834–2839, 2010.

[5] R. Ramadour, F. Chaumette, and J. P. Merlet, "Grasping objects with a cable-driven parallel robot designed for transfer operation by visual servoing," in *Proceedings - IEEE International Conference on Robotics and Automation*, Hong Kong, pp. 4463–4468, 2014.

[6] T. Dallej, M. Gouttefarde, N. Andreff, P. E. Hervé, and P. Martinet, "Modeling and vision-based control of large-dimension cable-driven parallel robots using a multiple-camera setup," *Mechatronics*, vol. 61, pp. 20–36, 2019.

[7] R. Chellal, L. Cuvillon, and E. Laroche, "A Kinematic Vision-Based Position Control of a 6-DoF Cable-Driven Parallel Robot," in *International Conference on Cable-Driven Parallel Robots*, pp. 213–225, 2015.

[8] Z. Zake, F. Chaumette, N. Pedemonte, and S. Caro, "Vision-based control and stability analysis of a cable-driven parallel robot," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1029–1036, 2019.

[9] Z. Zake, F. Chaumette, N. Pedemonte, and S. Caro, "Robust $21/2$d visual servoing of a cable-driven parallel robot thanks to trajectory tracking," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 660–667, 2020.

[10] M. Carricato and J. P. Merlet, "Stability analysis of underconstrained cable-driven parallel robots," *IEEE Transactions on Robotics*, vol. 29, no. 1, pp. 288–296, 2013.

[11] M. Carricato, "Direct Geometrico-Static Problem of Underconstrained Cable-Driven Parallel Robots with Three Cables," *Mechanisms and Machine Science*, vol. 5, no. 3, pp. 269–285, 2013.

[12] M. Carricato and G. Abbasnejad, "Direct Geometrico-Static Analysis of Under-Constrained Cable-Driven Parallel Robots with 4 Cables," in *International Conference on Cable-Driven Parallel Robots*, pp. 269–285, 2013.

[13] G. Abbasnejad and M. Carricato, "Direct Geometrico-static Problem of Underconstrained Cable-Driven Parallel Robots With n Cables," *IEEE Transactions on Robotics*, vol. 31, no. 2, pp. 468–478, 2015.

[14] A. Berti, "Kinematics and statics of cable-driven parallel robots by interval-analysis-based methods," PhD thesis, Université de Nice-Sophia Antipolis, 2015.

[15] A. Berti, J. P. Merlet, and M. Carricato, "Solving the direct geometrico-static problem of underconstrained cable-driven parallel robots by interval analysis," *International Journal of Robotics Research*, vol. 35, no. 6, pp. 723–739, 2016.

[16] J. P. Merlet and J. Alexandre-Dit-Sandretto, "The forward kinematics of cable-driven parallel robots with sagging cables," in *International Conference on Cable-Driven Parallel Robots*, pp. 3–15, 2015.

[17] J. F. Collard and P. Cardou, "Computing the lowest equilibrium pose of a cable-suspended rigid body," *Optimization and Engineering*, vol. 14, no. 3, pp. 457–476, 2013.

[18] S. Fang, *Design, modeling and motion control of tendon based parallel manipulators*. VDI-Verlag, 2005.

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

[20] F. Chaumette and S. Hutchinson, "Visual Servoing and visual tracking," in *Handbook of Robotics*, B. Siciliano and O. Khatib, Eds. Springer, pp. 563–583, 2008.

[21] W. Khalil and E. Dombre, *Modeling, identification and control of robots*, Butterworth-Heinemann, 2004.

[22] E. Eade, "Lie groups for computer vision," 2014.

[23] J. P. Merlet, "Simulation of discrete-time controlled cable-driven parallel robots on a trajectory," *IEEE Transactions on Robotics*, vol. 33, no. 3, pp. 675–688, 2017.

[24] A. Pott, "Influence of pulley kinematics on cable-driven parallel robots," *Latest Advances in Robot Kinematics*, pp. 197–204, 2012.

[25] M. P. Vlasblom and R. L. Bosman, "Predicting the creep lifetime of hmpe mooring rope applications," in *OCEANS 2006*, IEEE, pp. 1–10, 2006.

[26] E. Olson, "AprilTag: A robust and flexible visual fiducial system," in *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 3400–3407, 2011.

[27] É. Marchand, F. Spindler, and F. Chaumette, "ViSP for visual servoing: A generic software platform with a wide class of robot control skills," *IEEE Robotics and Automation Magazine*, vol. 12, no. 4, pp. 40–52, 2005.

[28] S. Briot, S. Caro, and C. Germain, "Design procedure for a fast and accurate parallel manipulator," *Journal of Mechanisms and Robotics*, vol. 9, no. 6, 2017.