Redundancy Resolution integrated Model Predictive Control of CDPRs: Concept, Implementation and Experiments

João Cavalcanti Santos, Ahmed Chemori, Marc Gouttefarde

Redundancy Resolution Integrated Model Predictive Control of CDPRs: Concept, Implementation and Experiments

João C. Santos [1], Ahmed Chemori [1] and Marc Gouttefarde[1]

***Abstract*— This paper introduces a Model Predictive Control (MPC) strategy for fully-constrained Cable-Driven Parallel Robots. The main advantage of the proposed scheme lies in its ability to explicitly handle cable tension limits. Indeed, the cable tension distribution is performed as an integral part of the main control architecture. This characteristic significantly improves the safety of the system. Experimental results demonstrate this advantage addressing a typical pick-and-place task with two different scenarios: nominal cable tension limits and reduced maximum tension. Satisfactory tracking errors were obtained in the first scenario. In the second scenario, the desired trajectory escapes from the workspace defined by the new set of tension limits. The proposed MPC scheme is able to minimize the tracking errors without violating the tension limits. Satisfying results were also obtained regarding robustness against uncertainties on the payload mass.**

I. INTRODUCTION

Cable-Driven Parallel Robots (CDPRs) consist mainly of a mobile platform attached to a base by several cables. Each cable has one end attached to the platform and the other one wound on a winch drum. By changing the free lengths of the cables, the platform can be displaced. CDPRs can have a large workspace, an advantageous payload to mass ratio, and heavy payload capabilities. Thanks to these advantages, CDPRs can be relevant solutions in several applications [1]–[4]. Nevertheless, the use of cables leads to reduced stiffness and potential issues related to accuracy and vibrations. The winches of large-dimension CDPRs use gear trains with large reduction ratios, introducing backlash and friction. Additionally, cables can pull but not push on the mobile platform and cable sagging should be avoided. Therefore, the control of large-dimension CDPRs should address several issues inherent to these robots.

This work presents results obtained within the context of the EU project Hephaestus [5]–[7]. This project should deliver an industrial solution able to aid the construction and maintenance of building facades. Therefore, safety, accuracy and disturbance rejection capabilities are major concerns. Moreover, the proposed solutions should be compatible with industrial components and software.

Most of the previously published works addressing the control of CDPRs propose model-based control schemes, typically including a feedforward input based on the dynamics of the system [8]–[15]. In conjunction with computed torque control, PID linear feedback control can lead to acceptable results [16]–[18]. However, improved performances may be obtained with advanced control schemes. Several studies successfully implemented Sliding Mode Control (SMC) [9]–[14]. This method allies finite time convergence, simple implementation and robustness towards uncertainties. Nevertheless, chattering phenomenon remains an issue in recent and advanced SMC applications [12]. Other methods such as robust H_∞ control [19], [20] and Lyapunov-based controllers [21] were also implemented.

In the case of redundantly actuated CDPRs, there exists infinitely many cable tensions to generate a given mobile platform wrench. Several studies have addressed this issue, e.g. [22]–[26]. The influence of the cable tension distribution on the performance of the robot is particularly important for fully-constrained CDPRs, which is the case studied in the present paper. For instance, Jamshidifar and Khajepour proposed in [27] a tension distribution method that optimizes the CDPR stiffness.

In general, the control scheme of a redundantly actuated CDPR includes both a tension distribution method and a feedback control strategy. Alternatively, we introduced in [7] the use of MPC to make the tension distribution an integral part of the control strategy. Simulation results showed that this control architecture may lead to superior performance compared to other state-of-the-art alternatives. One of the main advantages of MPC lies in the fact that control limits may be explicitly handled [28]. This advantage is of particular interest in the context of the control of CDPRs. A minimum cable tension limit reflects the constraint that cables should be kept sufficiently tightened. A maximum cable tension is mainly related to the maximum load that may be safely applied to the mechanical components of the robot. The violation of this limit may lead to catastrophic consequences.

The control strategies proposed in the above discussed works [8]–[21] are not able to handle cable tension limits within the main controller. These constraints are considered after the computation of the desired wrench to be applied on the platform. This computation is performed by the main control strategy, which does not assess the feasibility of the desired wrench. Hence, safety becomes a major concern if cable tensions beyond the maximum admissible values are necessary. To overcome this problem, we propose an MPC-based control strategy.

The main contribution of this paper lies in the design,

This work was supported by the European Union's H2020 Program (H2020/2014-2020) under the grant agreement No. 732513 (Hephaestus project). This work has also been sponsored by the French government research program Investissements d'avenir through the Robotex Equipment of Excellence (ANR-10-EQPX-44)

[1] The authors are with LIRMM, University of Montpellier, CNRS, Montpellier, France {joao.cavalcanti-santos, ahmed.chemori, marc.gouttefarde}@lirmm.fr

implementation and experimentation of an MPC scheme able to explicitly handle cable tension limits as an integral part of the control strategy. To the best of our knowledge, these are the first experimental results obtained with an MPC strategy applied to a 6 Degrees-of-Freedom (DOF) CDPR. The proposed scheme is based on the authors' previous work [7], where only simulation results were presented. In order to cope with the real-time application and physical limitations of CDPRs, some improvements on the method presented in [7] were necessary. These improvements are described in the present paper. Experiments were performed with a fully-constrained 6 DOF CDPR driven by 8 cables. The experimental setup is shown in Fig. 1. Satisfying results are obtained for a typical pick-and-place task. In order to show the effectiveness of the proposed controller, the same trajectory tracking problem is constrained with a maximum tension drastically reduced. The desired trajectory escapes from the robot workspace. The proposed MPC is able to comply with this reduced maximum cable tension while minimizing the cartesian error between the actual and the desired trajectories. This is an important result related to the safety of the operation of CDPRs.

The proposed control strategy also reduces the variations of cable tensions. Tensions with large variations are not physically feasible. State-of-the-art tension distribution methods [22]–[26] do not consider this issue. Therefore, large cable tensions variations may occur, especially during the robot initialization where the actual cable tensions are not necessarily equal to the optimal cable tension distribution. The proposed MPC is not prone to this issue.

Additionally, robustness towards uncertainty on the payload is experimentally evaluated. A payload mass is suddenly added to the platform mass during a vertical motion. Satisfying results were obtained with additional payloads weighing up to 50% of the platform mass, which is a relevant test in the context of typical load-lifting CDPR applications.

In addition to the proposed MPC strategy, this paper also introduces an inner feedback control loop of the cable tensions where the desired tensions are the output of the MPC. It is a non-model-based strategy controlling motor velocities.

Previous studies addressing the implementation of MPC in the control of CDPRs are very few and recent. Katliar *et al.* proposed a nonlinear MPC for a motion simulator in [29]. Its performance was investigated through numerical simulations. To the best of our knowledge, no experimental results were published so far. The feasibility of the MPC implementation in real time was addressed taking into account solvers such as HPMPC [30] and qpOASES [31]. However, these solvers are not compatible with common industrial real-time environments. In the present paper, experimental results were obtained with solvers developed by the authors in the industrial real-time TwinCAT environment. As a motion simulator control, the focus in [29] is a set of desired accelerations and velocities. The present work is focused on trajectory tracking, which prioritize positioning accuracy.

The recent work [32] presents simulations and experimental results of an MPC scheme implemented in a hybrid cable-driven robot. However, the studied cable-driven system consists of a 2-DOF platform with planar movements. Only some of the cables are controlled in tension. The remaining cables are controlled in position. The actual tensions of these cables are not considered in the real-time controller and the proposed MPC strategy is used to control the vibration of the system. In the present work, the CDPR has six DOFs, eight cables, and each cable tension is controlled by the MPC scheme. The focus of the present paper is motion control, instead of vibration attenuation.

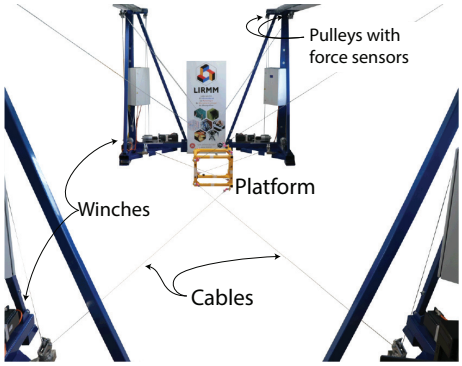

Fig. 1. View of HRPCable prototype.

The proposed MPC strategy was implemented in the industrial environment Beckhoff TwinCAT with a CDPR prototype built with components typically used in industry. The real-time implementation in TwinCAT is notably not compatible with common libraries of matrix manipulation. Therefore, the control scheme proposed in this paper was implemented from scratch. Given that the implementation of online optimization is an important constraint for real-time applications of constrained MPC schemes [33], details on the implementation of the numerical solutions are presented.

This paper is organized as follows. Section II briefly presents the kinematic and dynamic modeling of the CDPR. Section III introduces the control scheme. In Section IV, the obtained experimental results are presented and discussed. Conclusions are drawn in section V. Some notes on the numerical implementation are presented in the Appendix.

II. Kinematic and Dynamic Modeling

This section presents the model of a CDPR consisting of an n-DOF mobile platform driven by m cables, where $n \leq m$. Cable tensions $\boldsymbol{\tau}$ generate the platform wrench $\mathbf{f}$, where the wrench matrix $\mathbf{W}$ linearly maps $\boldsymbol{\tau} = \mathbf{W}\,\mathbf{f}$ [24]. Each length l_i is defined as the distance between the drawing point A_i and the attachment point B_i on the platform. The point A_i is the drawing point defined by the pulley attached to the base frame. The position of this point is considered constant. The cable length vector is $\boldsymbol{l} = \begin{bmatrix} l_1 & \dots & l_m \end{bmatrix}^T$. The cables are considered as being massless and their elasticity is neglected.

The platform pose is composed of the position and orientation vectors: $\mathbf{x} = [\mathbf{p}^T \ \boldsymbol{\psi}^T]^T$. Typically, $\boldsymbol{\psi}$ consists of Euler

angles. The vectors of platform velocity and acceleration are denoted by $\dot{\mathbf{x}}$ and $\ddot{\mathbf{x}}$, respectively[1].

The application of Newton-Euler formalism leads to the dynamics of the platform as [34]

$$\mathbf{M}(\mathbf{x})\,\ddot{\mathbf{x}} + \mathbf{C}(\mathbf{x},\dot{\mathbf{x}})\,\dot{\mathbf{x}} = \mathbf{g}(\mathbf{x}) + \mathbf{f} \tag{1}$$

where matrices $\mathbf{M}$ and $\mathbf{C}$ are given by

$$\mathbf{M}(\mathbf{x}) = \begin{bmatrix} m_p\,\boldsymbol{I}_3 & -m_p\,\hat{\mathbf{c}} \\ m_p\,\hat{\mathbf{c}} & \mathbf{H} \end{bmatrix}, \mathbf{C}(\mathbf{x},\dot{\mathbf{x}})\,\dot{\mathbf{x}} = \begin{bmatrix} m_p\,\hat{\boldsymbol{\omega}}\,\hat{\boldsymbol{\omega}}\,\mathbf{c} \\ \hat{\boldsymbol{\omega}}\,\mathbf{H}\,\boldsymbol{\omega} \end{bmatrix} \tag{2}$$

The scalar m_p denotes the platform mass and $\boldsymbol{I}_N$ is the identity matrix of dimension N. $\mathbf{c} = \begin{bmatrix} c_x & c_y & c_z \end{bmatrix}^T$ is the vector going from the platform geometric center to its center of mass. Matrices $\hat{\boldsymbol{\omega}}$ and $\hat{\mathbf{c}}$ are the skew-symmetric matrices associated to $\boldsymbol{\omega}$ and $\mathbf{c}$, respectively, with $\boldsymbol{\omega}$ the angular velocity of the platform. The matrix $\mathbf{H}$ is defined as $\mathbf{H} = \mathbf{I} + m_p\,\hat{\mathbf{c}}\,\hat{\mathbf{c}}^T$ where $\mathbf{I}$ is the platform inertia matrix relative to its center of mass. The vector of gravitational forces is given by $\mathbf{g}(\mathbf{x}) = m_p\,g\,\begin{bmatrix} 0 & 0 & -1 & -c_y & c_x & 0 \end{bmatrix}^T$.

No external measurement system is used in this work to estimate the CDPR mobile platform pose. Therefore, the pose and velocity of the platform are estimated based on the positions and velocities of the motors. Positions and velocities of the motors provide the values of $\boldsymbol{l}$ and $\dot{\boldsymbol{l}}$. Since the number of cables is greater than number of platform DOFs ($m \geq n$), the forward kinematic problem is over constrained. Consequently, the pose of the platform can be calculated as the vector $\mathbf{x}$ minimizing the difference between the lengths calculated from the motor angular positions and those consistent with the kinematic model. This method is presented in [35] and briefly discussed in the Appendix.

III. Control Design

The proposed control scheme is introduced in this section. The MPC is introduced in III-A and the cable tension control in III-B. The overall control scheme is illustrated in the block diagram of Fig. 2.

A. Model Predictive Control (MPC)

The proposed control scheme is based on [7]. Some modifications are introduced and discussed hereafter. Consider the discrete-time system with time-step Δt

$$\mathbf{y}(t+\Delta t) = \mathbf{A}(t)\,\mathbf{y}(t) + \mathbf{B}(t)\,\boldsymbol{\tau}(t) + \mathbf{v}(t) \tag{3}$$

which represents an approximation of the continuous model (1) with $\mathbf{y}(t+\Delta t) = \begin{bmatrix} \mathbf{x}(t+\Delta t)^T & \dot{\mathbf{x}}(t+\Delta t)^T \end{bmatrix}^T$. Matrices $\mathbf{v}$, $\mathbf{A}$ and $\mathbf{B}$ are functions of $\mathbf{x}$ and $\dot{\mathbf{x}}$. These matrices are defined in [7].

For a given sequence of cable tensions $\{\boldsymbol{\tau}(t), ..., \boldsymbol{\tau}\,\big(\,t + (h_p - 1)\,\Delta t\big)\}$, the recursive application of (3) leads to a sequence of states $\{\mathbf{y}(t+\Delta t), ..., \mathbf{y}(t+h_p\,\Delta t)\}$ within an horizon $\{t+\Delta t, ..., t+h_p\Delta t\}$, where the positive integer h_p denotes the *prediction horizon*. The proposed MPC scheme finds the optimum sequence of cable tensions within the prediction horizon with respect to a cost function J.

[1]Since angular velocity is not equal to the time derivative of the vector of Euler angles, it is worth to note that $\dot{\mathbf{x}} = \mathbf{S}(\boldsymbol{\psi})\,d\mathbf{x}/dt$, with an orientation-dependent matrix $\mathbf{S}$. Similarly, $\ddot{\mathbf{x}} = \mathbf{S}\,d^2\mathbf{x}/dt + \dot{\mathbf{S}}\,d\mathbf{x}/dt$, with $\dot{\mathbf{S}}$ the element-wise time derivative of $\mathbf{S}$. Moreover, in general, the matrix $-\mathbf{W}^T$ does not map velocities $d\mathbf{x}/dt$ into $\dot{\boldsymbol{l}}$.

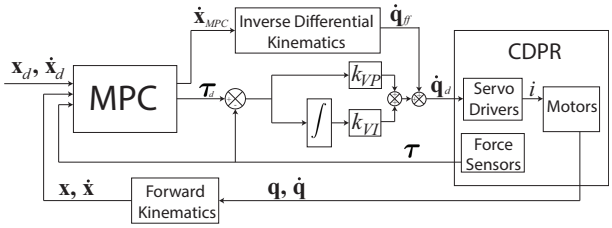

Fig. 2. Overview of the proposed control scheme

The matrices $\mathbf{v}$, $\mathbf{A}$ and $\mathbf{B}$ are function of the pose and velocity of the platform. Considering that the interval of time $h_p\Delta t$ is sufficiently small and assuming low velocities, the latter assumption being valid since this work focuses on large-dimension CDPRs moving at low velocities, the variation of these matrices may be disregarded. The linear time-varying system (3) is thus approximated as a linear time-invariant system over the prediction horizon, i.e., $\mathbf{A}(t + i\,\Delta t) = \mathbf{A}(t)$, $\mathbf{B}(t+i\,\Delta t) = \mathbf{B}(t)$ and $\mathbf{v}(t+i\,\Delta t) = \mathbf{v}(t)$ for $i = 1, ..., h_p$.

The proposed MPC scheme considers different values of $\boldsymbol{\tau}$ for the first h_c vectors of cable tensions over the prediction horizon, with $h_c \leqslant h_p$. The positive integer h_c is called the *control horizon*. The remaining vectors of cable tensions are considered equal to the last one, i.e., $\boldsymbol{\tau}(t + k\,\Delta t) = \boldsymbol{\tau}\big(t + (h_c - 1)\Delta t\big)$ for $h_c \leqslant k \leqslant h_p$. The vectors of future predicted states, desired states and cable tensions are, respectively, $\mathbf{Y}(t) = \begin{bmatrix} \mathbf{y}(t+\Delta t)^T & ... & \mathbf{y}(t+h_p\,\Delta t)^T \end{bmatrix}^T$, $\mathbf{Y}_d(t) = \begin{bmatrix} \mathbf{y}_d(t+\Delta t)^T & \dots & \mathbf{y}_d(t+h_p\,\Delta t) \end{bmatrix}^T$ and $\mathbf{U}(t) = \begin{bmatrix} \boldsymbol{\tau}(t)^T & ... & \boldsymbol{\tau}(t+(h_c-1)\,\Delta t)^T \end{bmatrix}^T$, where $\mathbf{y}_d(t) = \begin{bmatrix} \mathbf{x}_d(t)^T & \dot{\mathbf{x}}_d(t)^T \end{bmatrix}^T$ is the desired state at time t. The lengths of the vectors $\mathbf{Y}$, $\mathbf{Y}_d$ and $\mathbf{U}$ are $2\,n\,h_p$, $2\,n\,h_p$ and $m\,h_c$, respectively. Future states are derived applying (3) recursively, which leads, over the prediction horizon, to

$$\mathbf{Y}(t) = \mathbf{D}(t)\,\mathbf{y}(t) + \mathbf{E}(t)\,\mathbf{U}(t) + \mathbf{F}(t) \tag{4}$$

with $\mathbf{D}(t)$ and $\mathbf{F}(t)$ defined as in [7]. These matrices are updated every sampling time. Matrix $\mathbf{E}$ is given by

$$\mathbf{E} = \begin{bmatrix} \mathbf{B} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{A}\mathbf{B} & \mathbf{B} & 0 & \vdots \\ \mathbf{A}^2\mathbf{B} & \mathbf{A}\mathbf{B} & \ddots & \mathbf{0} \\ \vdots & \vdots & & \vdots \\ \mathbf{A}^{h_c}\mathbf{B} & \mathbf{A}^{h_c-1}\mathbf{B} & \dots & \mathbf{B} \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{A}^{h_p-1}\mathbf{B} & ... & \mathbf{A}^{h_p-h_c+1}\mathbf{B} & \sum_{i=0}^{h_p-h_c} \mathbf{A}^i\,\mathbf{B} \end{bmatrix}$$

where the dependence on t was omitted and the size of the matrix $\mathbf{E}$ is $2\,n\,h_p \times m\,h_c$.

Better performance is obtained with the proposed MPC scheme for larger values of h_p and h_c. Nevertheless, in real-time applications, these values are constrained due to computation time limits. The influence of h_c on the computation time is larger than the influence of h_p. Therefore, a small reduction of h_c may lead to an important augmentation of h_p. For this reason, whereas simulations results presented in [7] consider $h_p = h_c$ (matrix $\mathbf{E}$ as in [7]), the present work applies $h_c < h_p$. This is the first improvement applied to the MPC scheme of [7].

The cost function considered by the MPC scheme includes (i) the predicted errors, (ii) the magnitude of the cable tensions and (iii) the variation of the cable tensions. Simulations in [7] include (i) and (ii), but do not include (iii). This is the main difference between the control scheme used in [7] and the present one. The minimization of the cable tensions variation (iii) is important mainly because large variations of cable tensions are not physically feasible. This issue is also relevant during the initialization of the robot. If the initial cable tensions are not close to an optimum cable tension distribution, the MPC will automatically find a smooth transition between these sets of cable tensions. Hence, the inclusion of (iii) to the cost function J represents the second improvement applied to the MPC scheme proposed in [7]. Weighted expressions of (i), (ii) and (iii) may be respectively given as follows

$$J_Y(\mathbf{y}, \mathbf{U}) = (\mathbf{Y}_d - \mathbf{Y})^T \mathbf{K}_Y (\mathbf{Y}_d - \mathbf{Y}) \tag{5}$$

$$J_U(\mathbf{U}) = \mathbf{U}^T \mathbf{K}_U \mathbf{U} \tag{6}$$

$$J_D(\mathbf{U}) = k_{\Delta u} \sum_{i=0}^{hc-1} \boldsymbol{\Delta}\mathbf{u}(t + i\,\Delta u)^T\, \boldsymbol{\Delta}\mathbf{u}(t + i\,\Delta t) \tag{7}$$

where $\mathbf{K}_Y$ and $\mathbf{K}_U$ are positive definite diagonal weighting matrices and $k_{\Delta u}$ is a positive scalar. The vector of the variation of cable tensions can be expressed as

$$\boldsymbol{\Delta}\mathbf{u}(t + i\,\Delta t) = \boldsymbol{\tau}(t + i\Delta t) - \boldsymbol{\tau}\big(t + (i-1)\Delta t\big) \tag{8}$$

Note that the variations of cable tensions are given by

$$\boldsymbol{\Delta}\mathbf{U}(t) = \mathbf{Q}\,\mathbf{U}(t) - \mathbf{z}(t) \tag{9}$$

with

$$\mathbf{Q} = \begin{bmatrix} \boldsymbol{I}_m & \mathbf{0} & \dots & & \\ -\boldsymbol{I}_m & \boldsymbol{I}_m & \mathbf{0} & \dots & \\ \mathbf{0} & -\boldsymbol{I}_m & \boldsymbol{I}_m & \dots & \\ \vdots & \ddots & \ddots & \ddots & \mathbf{0} \\ \mathbf{0} & \dots & \dots & -\boldsymbol{I}_m & \boldsymbol{I}_m \end{bmatrix},\ \mathbf{z}(t) = \begin{bmatrix} \boldsymbol{\tau}(t - \Delta t) \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix}$$

and $\boldsymbol{\Delta}\mathbf{U}(t) = \big[\boldsymbol{\Delta}\mathbf{u}(t)^T \ \dots \ \boldsymbol{\Delta}\mathbf{u}(t + (h_c - 1)\,\Delta t)^T\big]^T$. Note that $\mathbf{z}$ depends on the measured cable tensions. Alternatively, the desired cable tension obtained for $t - \Delta t$ may be used. The sizes of $\mathbf{Q}$ and $\mathbf{z}$ are $m\,h_c \times m\,h_c$ and $m\,h_c \times 1$, respectively.

Defining $\mathbf{K}_D = k_{\Delta u}\,\boldsymbol{I}_{h_c.m}$, J_D can be rewritten as

$$J_D = \mathbf{U}^T \mathbf{Q}^T \mathbf{K}_D\, \mathbf{Q}\,\mathbf{U} - 2\,\mathbf{z}^T \mathbf{K}_D \mathbf{Q}\,\mathbf{U} + \mathbf{z}^T \mathbf{K}_D\, \mathbf{z}. \tag{10}$$

Using (4), the overall cost function $J(\mathbf{y}, \mathbf{U})$ is defined as the sum of J_Y, J_U and J_D disregarding the terms independent of $\mathbf{U}$:

$$J(\mathbf{y}, \mathbf{U}) = \mathbf{U}^T \overbrace{\left(\mathbf{E}^T \mathbf{K}_Y\, \mathbf{E} + \mathbf{K}_U + \mathbf{Q}^T \mathbf{K}_D\, \mathbf{Q}\right)}^{\mathbf{H}_c} \mathbf{U} + \\ + 2 \underbrace{\left((\mathbf{D}\,\mathbf{y} + \mathbf{F} - \mathbf{Y}_d)^T \mathbf{K}_Y\, \mathbf{E} - \mathbf{z}^T \mathbf{K}_D \mathbf{Q}\right)}_{\mathbf{d}^T} \mathbf{U}$$

The proposed MPC consists on computing the optimal sequence of cable tensions by solving the following Quadratic Programming problem in real-time

$$\begin{aligned} \mathbf{U}^* = \arg\min_{\mathbf{U}} \quad & \frac{1}{2}\mathbf{U}^T \mathbf{H}_c \mathbf{U} + \mathbf{d}^{\mathbf{T}} \mathbf{U} \\ \text{s.t.} \quad & \tau_{min} \leqslant \mathbf{U} \leqslant \tau_{max} \end{aligned} \tag{11}$$

where τ_{min} and τ_{max} are the minimum and maximum admissible cable tensions, respectively. The optimal solution $\mathbf{U}^*(t) = [\boldsymbol{\tau}_d^T(t),\ \boldsymbol{\tau}_d^T(t + \Delta t), ..., \boldsymbol{\tau}_d^T(t + (h_c - 1)\Delta t)]^T$ represents the optimal control inputs over the control horizon. The output of the MPC scheme is the next desired cable tension $\boldsymbol{\tau}_d(t)$. This procedure is repeated at each sampling time, updating matrices $\mathbf{H}_c$ and $\mathbf{d}$ in function of $\mathbf{y}$ and $\boldsymbol{\tau}$. The Cartesian velocity $\dot{\mathbf{x}}_{MPC}$ predicted by the MPC is used in the cable tension control. This vector is obtained with $\mathbf{y}(t + \Delta t) = \mathbf{A}(t)\,\mathbf{y}(t) + \mathbf{B}(t)\,\boldsymbol{\tau}_d(t) + \mathbf{v}(t)$.

B. Cable Tension Control

As illustrated in Fig. 2, the desired cable tensions $\boldsymbol{\tau}_d$ are computed by the proposed MPC scheme and form the input to the inner control loop. The CDPR winch motors are controlled in velocity. The desired motor velocity $\dot{\mathbf{q}}_d$ is obtained with a PI controller with a feedforward term, as follows

$$\dot{\mathbf{q}}_d(t) = P\left(\boldsymbol{\tau}_d(t) - \boldsymbol{\tau}(t)\right) + I \int_{t_0}^{t} \left(\boldsymbol{\tau}_d(t) - \boldsymbol{\tau}(t)\right) dt + \dot{\mathbf{q}}_{ff}$$

where $\dot{\mathbf{q}}_{ff}$ is the feedforward term obtained by applying the inverse differential kinematics with Cartesian velocity equals to $\dot{\mathbf{x}}_{MPC}$.

IV. Real-Time Experimental Results

A. Experimental platform and implementation details

The control scheme introduced in Section III is experimented on HRPCable, a 6-DOF CDPR driven by 8 cables, installed in LIRMM facilities. Figure 1 shows this experimental setup. The corresponding CAD model is shown in Fig. 4. The footprint of this robot is 10 x 4 x 3 (m) and, in the experiments reported in this paper, the cables are arranged so as to fully constrain the mobile platform. Winches are driven by motors Beckhoff AM8061. The servo drives AX5112 are driven through EtherCAT communication by an industrial PC C6920 equipped with 2.4GHz i7 core processor. The motor reduction is performed by a two stage gear train Beckhoff AG2210 with a reduction ratio of 25. Load pins Sensy 5300 1T SL positioned in the axes of the routing pulleys measure the cable tensions. The platform is a cube with length equal to 1 m and total mass equal to $m_p = 23$ kg.

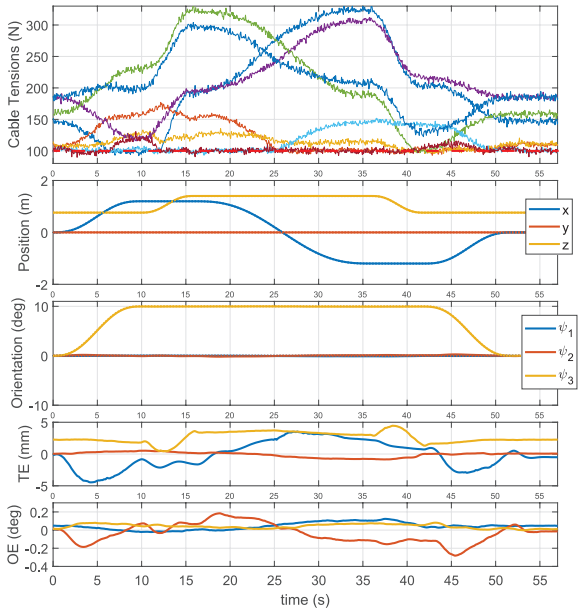

Fig. 3. Pick-and-place task results with $\tau_{max} = 400$ N. The actual pose is shown in continuous lines and the desired one in dashed lines.

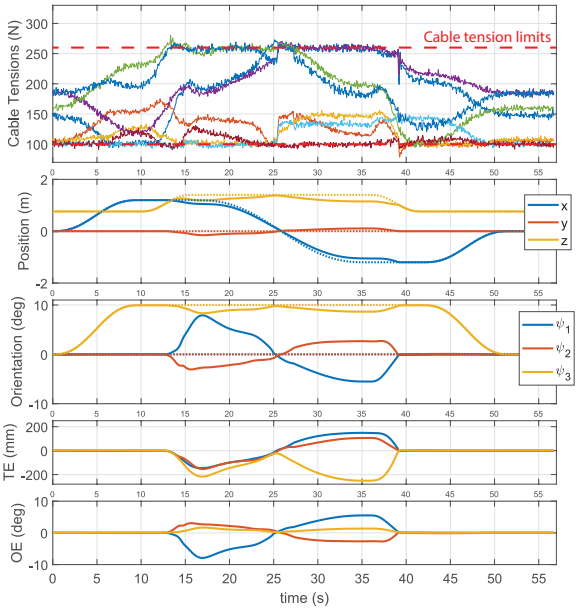

Fig. 5. Pick-and-place task results with $\tau_{max} = 260$ N. the actual pose is depicted in continuous lines and the desired pose in dashed lines.

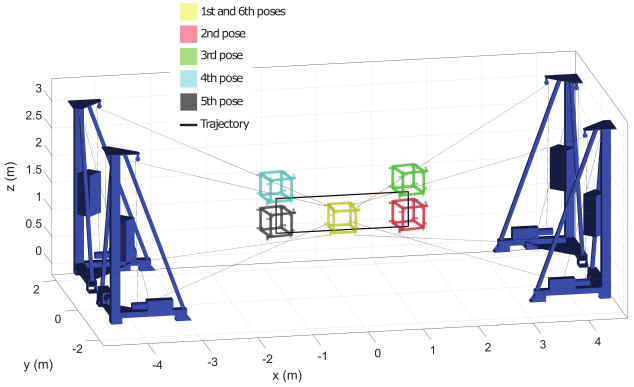

Fig. 4. Pick-and-place path and CAD view of HRPCable.

The control scheme introduced in Section III has been implemented in an industrial PC and developed in TwinCAT, using C++ language. TwinCAT is a software from Beckhoff commonly used in the industry which turns PC-based systems into real-time controllers using Microsoft Windows kernel. In order to guarantee real-time performances, TwinCAT is not compatible with standard C++ libraries, including basic libraries such as `math.h`. This issue generally mitigates the application of MPC with constraints in industrial robotics. The control methods described in the previous sections were thus programmed from scratch. Some notes on the numerical implementation are presented in the Appendix.

In the experiments reported below, the control parameters are the following: $h_p = 6$, $h_c = 3$, $\mathbf{K}_U = 3.3 \times 10^{-3} \boldsymbol{I}_{h_c.m}$, $k_{\Delta u} = 0.3$. Matrix $\mathbf{K}_Y$ is diagonal and positive definite. Diagonal elements are repeated for every set of 12 elements (size of $\mathbf{y}$). These 12 elements are equal to 2×10^7, 2×10^7, 3.3×10^7, 2×10^7, 2×10^7, 2×10^7, 1.2×10^{-3}, 0.6×10^{-3}, 0.6×10^{-3}, 0.6×10^{-3}, 1.3×10^{-3}, 0.6×10^{-3}. These gains have been tuned manually.

B. Pick-and-Place task

The first scenario considered is a typical pick-and-place task. Figure 4 shows the sequence of the six desired platform poses defining the pick-and-place trajectory. The trajectory between each subsequent pair of desired poses is defined with a 5^{th} degree polynomial. In the following, the results obtained with nominal cable tension constraints and with reduced τ_{max} are compared. The reader is invited to watch the attached video, which compares these two cases.

1) Nominal Constraints: Figure 3 shows the experimental results obtained with $\tau_{max} = 400$ N and $\tau_{min} = 100$ N. The proposed controller is able to keep the translation errors (TE) smaller than 4.5 mm and the orientation errors (OE) smaller than $0.3°$.

2) Reduced Maximum Tension: Figure 5 shows the experimental results obtained with $\tau_{max} = 260$ N and $\tau_{min} = 100$ N. The proposed MPC finds poses as close as possible to desired poses complying with these cable tension limits. Translation errors reach values greater than 250 mm and orientation errors are close to $8°$. Strictly complying with the cable tension limits while minimizing the Cartesian errors in following the desired trajectory is the main advantage of the proposed controller. The control strategies proposed in [8]–[21] combined with some of the tension distribution schemes in [22]–[27] would fail to fulfill this objective since the main controller would demand an unfeasible wrench.

C. Robustness against payload uncertainties

In many applications of CDPRs, the platform should pick and release weights. In order to validate the applicability of

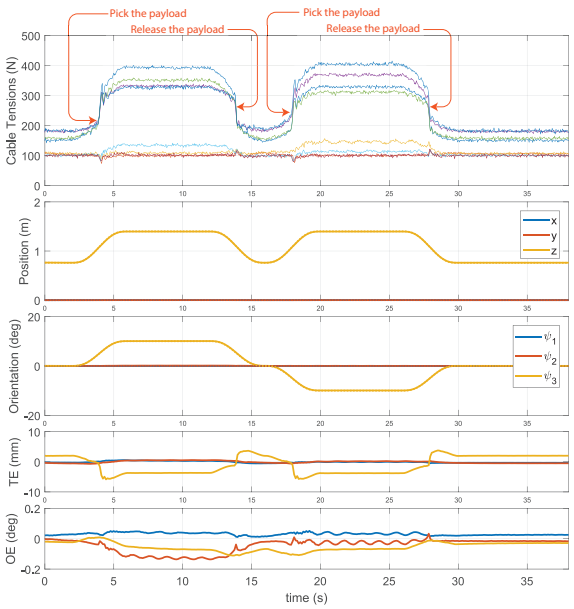

Fig. 6. Robustness test results with an additional payload.

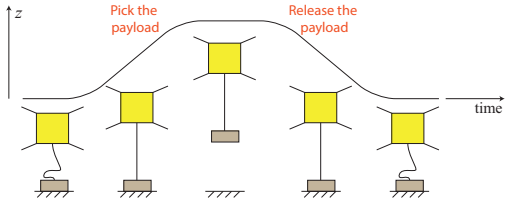

Fig. 7. Illustration of the robustness test.

the proposed control strategy in such tasks, it is important to evaluate its robustness against uncertainties on the lifted mass. To this end, the experiment illustrated in Fig. 7 is proposed. The results are presented in Fig. 6. An additional mass of 11.5 kg was used. This mass represents 50% of the platform mass m_p. The controller is able to keep the tracking errors smaller than 6 mm and 0.14°, despite this uncertainty.

V. Conclusion

This paper introduces an MPC scheme for CDPRs. Experimental results on a fully-constrained 6-DOF CDPR show that the proposed control strategy is able to perform a trajectory keeping reduced errors while complying with cable tension limits. The real-time control was implemented in an industrial software and hardware environment to enable applicability of the proposed scheme in industry. The proposed MPC is able to address the cable tension limits explicitly, integrating the redundancy resolution within the main controller. As a result, cable tension limits are not violated even for reduced maximum cable tensions. Indeed, when a desired pose cannot be reached with some given tension limits, the proposed MPC is able to find a trajectory as close as possible to the desired one while strictly respecting the tension limits. Experiments also show its robustness against payload uncertainties.

APPENDIX

An important part of the present work lies in the numerical implementation of the proposed control strategy in a real-time industrial software environment. This appendix addresses some details of this implementation.

The Forward Kinematic algorithm used in this work is slightly different from the state-of-the-art published methods. Indeed, QR decomposition are used to iteratively solve the linearized version of the Forward Kinematic problem as described in [35].

Regarding the solution of (11), an adapted Active Set method is used [36]. The Active Set method solves a simplified version of (11) taking a subset of the constraints as *equality constraints* at each iteration (see [36] for further details). The resulting subproblem in a given iteration can be written as

$$\begin{aligned} \min_{\mathbf{U}} \quad & \frac{1}{2}\mathbf{U}^T\mathbf{H}_c\mathbf{U} + \mathbf{d}^\mathbf{T}\mathbf{U} \\ \text{s.t.} \quad & \begin{cases} u_i = \tau_{min}, & \text{for } i \in \mathcal{M} \\ u_j = \tau_{max}, & \text{for } j \in \mathcal{N} \end{cases} \end{aligned} \tag{12}$$

with $\mathcal{M} \subset \{1, ..., m\,h_c\}$, $\mathcal{N} \subset \{1, ..., m\,h_c\}$ and $\mathcal{M} \cap \mathcal{N} = \varnothing$. Take $\mathcal{A} = (\mathcal{M} \cup \mathcal{N})$ and $\mathcal{F} = \{1, ..., m\,h_c\} - (\mathcal{M} \cup \mathcal{N})$. For notation simplicity, let us consider that $\mathcal{F} = \{1, ..., n_f\}$, $\mathcal{M} = \{n_f + 1, \ldots, n_f + n_m\}$ and $\mathcal{N} = \{h_c\,m - n_n + 1, \ldots, h_c\,m\}$, with n_f the number of elements of $\mathcal{F}$, n_m the number of elements of $\mathcal{M}$ and n_n the number of elements of $\mathcal{N}$. $\mathbf{H}_c$, $\mathbf{U}$ and $\mathbf{d}$ in (12) can be written as follows

$$\mathbf{H_c} = \begin{bmatrix} \mathbf{H}_f & \mathbf{H}_1 \\ \mathbf{H}_1^T & \mathbf{H}_2 \end{bmatrix}; \quad \mathbf{U} = \begin{bmatrix} \mathbf{U}_f \\ \mathbf{U}_m \\ \mathbf{U}_n \end{bmatrix}; \quad \mathbf{d} = \begin{bmatrix} \mathbf{d}_f \\ \mathbf{d}_m \\ \mathbf{d}_n \end{bmatrix}$$

where $\text{size}(\mathbf{H}_f) = (n_f, n_f)$, $\text{size}(\mathbf{H}_1) = (n_f, n_m + n_n)$, $\text{size}(\mathbf{H}_2) = (n_m + n_n, n_m + n_n)$, $\text{size}(\mathbf{U}_f) = (n_f, 1)$, $\text{size}(\mathbf{U}_m) = (n_m, 1)$ and $\text{size}(\mathbf{U}_n) = (n_n, 1)$. Note that each element of $\mathbf{U}_m$ is equal to τ_{min} and each element of $\mathbf{U}_n$ is equal to τ_{max}. The solution of (12) is $(\mathbf{U}^*)^T = \begin{bmatrix}(\mathbf{U}_f^*)^T & \mathbf{U}_m^T & \mathbf{U}_n^T\end{bmatrix}$, with $\mathbf{U}_f^*$ the vector that solves $\mathbf{H}_f\,\mathbf{U}_f = -\mathbf{d}_f - \mathbf{H}_1\begin{bmatrix}\mathbf{U}_m^T & \mathbf{U}_n^T\end{bmatrix}^T$.

It is worth noting that $\mathbf{H}_f$ is symmetric and positive definite. As a consequence, $\mathbf{H}_f$ may be factorized with Cholesky decomposition such that $\mathbf{H}_f = \mathbf{L}\,\mathbf{L}^T$, with $\mathbf{L}$ lower triangular. Cholesky decomposition is about a factor of two faster than alternative methods for solving linear equations and is extremely stable numerically [37]. The optimum is obtained from two successive back substitution procedures: $\mathbf{L}\,\mathbf{s} = -\mathbf{d}_f - \mathbf{H}_1\begin{bmatrix}\mathbf{U}_m^T & \mathbf{U}_n^T\end{bmatrix}^T$, and then $\mathbf{L}^T\mathbf{U} = \mathbf{s}$.

Before starting the next iteration of the Active Set method, components of $\mathbf{U}_f^*$ should comply with cable tension limits. Each element $\overline{u}_i$ of the vector $\overline{\mathbf{U}}$ used in the next iteration is obtained as a saturation of the corresponding element u_i^* of $\mathbf{U}^*$

$$\overline{u}_i = \min\left(\max(u_i^*, \tau_{min}), \tau_{max}\right) \tag{13}$$

The vector $\overline{\mathbf{U}}$ is used as input for the next iteration of the Active Set method changing the set of active constraints, as described in [36].

REFERENCES

[1] J. Albus, R. Bostelman, and N. Dagalakis, "The NIST ROBOCRANE," *J. Robotic Syst.*, vol. 10, no. 5, pp. 709–724, 1993.

[2] P. Miermeister, W. Kraus, T. Lan, and A. Pott, "An elastic cable model for cable-driven parallel robots including hysteresis effects," in *Mechanisms and Machine Science*. Springer, 2015, vol. 32, pp. 17–28.

[3] S. Kawamura, H. Kino, and C. Won, "High-speed manipulation by using parallel wire-driven robots," *Robotica*, vol. 18, no. 1, pp. 13–21, 2000. [Online]. Available: http://www.journals.cambridge.org/abstract_S0263574799002477

[4] C. Lambert, M. Nahon, and D. Chalmers, "Implementation of an aerostat positioning system with cable control," *IEEE/ASME Transactions on Mechatronics*, vol. 12, no. 1, pp. 32–40, 2007.

[5] European Union. (2017) Hephaestus project. [Online]. Available: www.hephaestus-project.eu

[6] H. Hussein, J. C. Santos, and M. Gouttefarde, "Geometric Optimization of a Large Scale CDPR Operating on a Building Facade," in *IEEE International Conference on Intelligent Robots and Systems*, 2018, pp. 5117–5124.

[7] J. C. Santos, A. Chemori, and M. Gouttefarde, "Model Predictive Control of Large-Dimension Cable-Driven Parallel Robots," in *International Conference on Cable-Driven Parallel Robots*. Springer, 2019, pp. 221–232.

[8] R. L. Williams, P. Gallina, and J. Vadia, "Planar translational cable-direct-driven robots," *Journal of Robotic Systems*, vol. 20, no. 3, pp. 107–120, 2003.

[9] A. Alikhani and M. Vali, "Sliding Mode Control of a Cable-driven Robot via Double-Integrator Sliding Surface," in *2012 International Conference on Control, Robotics, and Cybernetics*, 2012.

[10] G. El-Ghazaly, M. Gouttefarde, and V. Creuze, "Adaptive terminal sliding mode control of a redundantly-actuated cable-driven parallel manipulator: CoGiRo," in *Mechanisms and Machine Science*. Springer, 2015, vol. 32, pp. 179–200.

[11] W. Lv, L. Tao, and Z. Ji, "Sliding Mode Control of Cable-Driven Redundancy Parallel Robot with 6 DOF Based on Cable-Length Sensor Feedback," *Mathematical Problems in Engineering*, vol. 2017, 2017.

[12] C. Schenk, C. Masone, A. Pott, and H. H. Bülthoff, "Application of a differentiator-based adaptive super-twisting controller for a redundant cable-driven parallel robot," in *Mechanisms and Machine Science*. Springer, 2018, vol. 53, pp. 254–267.

[13] M. H. Korayem, H. Tourajizadeh, M. Jalali, and E. Omidi, "Optimal path planning of spatial cable robot using optimal sliding mode control," *International Journal of Advanced Robotic Systems*, vol. 9, no. 5, p. 168, 2012.

[14] M. Zeinali and A. Khajepour, "Design and Application of Chattering-Free Sliding Mode Controller to Cable-Driven Parallel Robot Manipulator: Theory and Experiment," in *Volume 2: 34th Annual Mechanisms and Robotics Conference, Parts A and B*. ASME, January 2010, pp. 319–327.

[15] J. Lamaury and M. Gouttefarde, "Control of a large redundantly actuated cable-suspended parallel robot," in *2013 IEEE International Conference on Robotics and Automation*, May 2013, pp. 4659–4664.

[16] M. H. Korayem, H. Tourajizadeh, and M. Bamdad, "Dynamic load carrying capacity of flexible cable suspended robot: Robust feedback linearization control approach," *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 60, no. 3-4, pp. 341–363, 2010.

[17] M. A. Khosravi and H. D. Taghirad, "Robust PID control of fully-constrained cable driven parallel robots," *Mechatronics*, vol. 24, no. 2, pp. 87 – 97, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0957415813002353

[18] M. A. Khosravi and H. D. Taghirad, "Dynamic modeling and control of parallel robots with elastic cables: Singular perturbation approach," *IEEE Transactions on Robotics*, vol. 30, no. 3, pp. 694–704, June 2014.

[19] R. Chellal, L. Cuvillon, and E. Laroche, "Model identification and vision-based H-infinity position control of 6-dof cable-driven parallel robots," *International Journal of Control*, vol. 90, no. 4, pp. 684–701, 2017. [Online]. Available: https://doi.org/10.1080/00207179.2016.1220623

[20] E. Laroche, R. Chellal, L. Cuvillon, and J. Gangloff, "A preliminary study for H-infinity control of parallel cable-driven manipulators," in *Cable-Driven Parallel Robots*. Springer, 2013, pp. 353–369.

[21] A. B. Alp and S. K. Agrawal, "Cable suspended robots: design, planning and control," in *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No.02CH37292)*, vol. 4, May 2002, pp. 4275–4280 vol.4.

[22] C. M. Gosselin and M. Grenier, "On the determination of the force distribution in overconstrained cable-driven parallel mechanisms," *Meccanica*, vol. 46, no. 1, pp. 3–15, 2011.

[23] A. Pott, "An improved force distribution algorithm for over-constrained cable-driven parallel robots," in *Mechanisms and Machine Science*. Springer, 2014, vol. 15, pp. 139–146.

[24] M. Gouttefarde, J. Lamaury, C. Reichert, and T. Bruckmann, "A Versatile Tension Distribution Algorithm for n-DOF Parallel Robots Driven by n + 2 Cables," *IEEE Transactions on Robotics*, vol. 31, no. 6, pp. 1444–1457, 2015.

[25] H. Yuan, E. Courteille, and D. Deblaise, "Force Distribution With Pose-Dependent Force Boundaries for Redundantly Actuated Cable-Driven Parallel Robots," *Journal of Mechanisms and Robotics*, vol. 8, no. 4, 2016.

[26] H. D. Taghirad and Y. B. Bedoustani, "An analytic-iterative redundancy resolution scheme for cable-driven redundant parallel manipulators," *IEEE Transactions on Robotics*, vol. 27, no. 6, pp. 1137–1143, Dec 2011.

[27] H. Jamshidifar, A. Khajepour, B. Fidan, and M. Rushton, "Kinematically-constrained redundant cable-driven parallel robots: Modeling, redundancy analysis, and stiffness optimization," *IEEE/ASME Transactions on Mechatronics*, vol. 22, no. 2, pp. 921–930, April 2017.

[28] J. M. Maciejowski, *Predictive control: with constraints*. Pearson education, 2002.

[29] M. Katliar, J. Fischer, G. Frison, M. Diehl, H. Teufel, and H. H. Blthoff, "Nonlinear model predictive control of a cable-robot-based motion simulator," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 9833 – 9839, 2017, 20th IFAC World Congress. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S2405896317313678

[30] G. Frison, H. H. B. Srensen, B. Dammann, and J. B. Jrgensen, "High-performance small-scale solvers for linear model predictive control," in *2014 European Control Conference (ECC)*, June 2014, pp. 128–133.

[31] H. J. Ferreau, C. Kirches, A. Potschka, H. G. Bock, and M. Diehl, "qpOASES: a parametric active-set algorithm for quadratic programming," *Mathematical Programming Computation*, vol. 6, no. 4, pp. 327–363, Dec 2014. [Online]. Available: https://doi.org/10.1007/s12532-014-0071-1

[32] R. Qi, M. Rushton, A. Khajepour, and W. W. Melek, "Decoupled modeling and model predictive control of a hybrid cable-driven robot (HCDR)," *Robotics and Autonomous Systems*, vol. 118, pp. 1–12, 2019.

[33] Y. Wang and S. Boyd, "Fast model predictive control using online optimization," *IEEE Transactions on Control Systems Technology*, vol. 18, no. 2, pp. 267–278, March 2010.

[34] J. Lamaury, M. Gouttefarde, A. Chemori, and P. E. Herve, "Dual-space adaptive control of redundantly actuated cable-driven parallel robots," in *IEEE International Conference on Intelligent Robots and Systems*. IEEE, 2013, pp. 4879–4886.

[35] J. C. Santos and M. Gouttefarde, "A real-time capable forward kinematics algorithm for cable-driven parallel robots considering pulley kinematics," in *Advances in Robot Kinematics 2020*. Springer, 2020, submitted for publication.

[36] J. Nocedal and S. Wright, *Numerical optimization*. Springer Science & Business Media, 2006.

[37] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, "Numerical recipes in C++," *The art of scientific computing*, vol. 2, p. 1002, 1992.