Continuous Tension Validation for Cable-Driven Parallel Robots

Diane Bury, Jean-Baptiste Izard, Marc Gouttefarde, Florent Lamiraux

Continuous Tension Validation for Cable-Driven Parallel Robots

Diane Bury[1,2], Jean-Baptiste Izard[1] , Marc Gouttefarde[3], Florent Lamiraux[2]

Abstract— **This paper deals with continuous tension validation for Cable-Driven Parallel Robots (CDPRs). The proposed method aims at determining whether or not a quasi-static path is feasible regarding cable tension limits. The available wrench set (AWS) is the set of wrenches that can be generated with cable tensions within given minimum and maximum limits. A pose of the robot is considered valid regarding the tensions if and only if the wrench induced by the platform weight is inside the AWS. The hyperplane shifting method gives a geometric representation of the AWS as the intersection of half-spaces. For each facet-defining hyperplane of the AWS, we define a value which is positive when the pose is valid, i.e. when the corresponding wrench lies on the proper side of the hyperplane. Using this value and an upper bound on its time derivative along the path, the half-length of a valid time interval is obtained. Intervals are repeatedly validated for each hyperplane until either the whole path is validated or a non-valid pose is found. The presented method is integrated within the open-source software Humanoid Path Planner (HPP) and implementation results using the configuration of the CDPR CoGiRo are presented.**
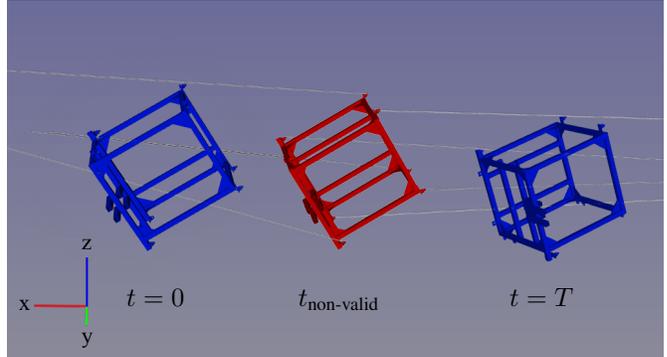
Fig. 1: Vizualisation of a path of the CDPR CoGiRo, parametrized by $[0, T]$. The first and last poses of the path are valid (blue). The continuous tension validation has found a non-valid pose (red) at parameter $t_{\text{non-valid}}$, therefore the path is not valid. Cables (grey) are shown only for the red pose.

I. INTRODUCTION

A cable-driven parallel robot (CDPR) is a parallel robot whose mobile platform is suspended by flexible cables, which serve as actuated limbs. By modifying the lengths of the cables, the position and orientation of the mobile platform can be controlled, as well as the wrench applied to it. CDPRs present the advantage of having large worskpaces compared to standard robots. However, the workspace of a CDPR is constrained by the limits on the cable tensions. Indeed, since cables can only pull and not push, their tension has to be positive. They also have a maximum allowed tension, which is set by the admissible loads on the robot mechanical components weighted by safety factors. The minimum and maximum tensions limit the reachable positions and the possible wrenches that can be generated at the platform. This paper deals with continuous cable tension validation for CD-PRs. The problem consists in determining exactly whether an input path between two poses (position and orientation) of the robot, which is subjected to the wrench induced by its weight (quasi-static motion), is feasible regarding cable tension limits, or if there exists a pose along the path for which there exists no feasible tension set.

[1]TECNALIA, Basque Research and Technology Alliance (BRTA), Bat 6 CSU, 950 rue Saint-Priest, 34090 Montpellier, France
`diane.bury@tecnalia.com`
`jeanbaptiste.izard@tecnalia.com`
[2]LAAS-CNRS, University of Toulouse, Toulouse, France
`florent.lamiraux@laas.fr`
[3]LIRMM, Université de Montpellier, CNRS, Montpellier, France
`marc.gouttefarde@lirmm.fr`

The method developed in this paper is applied to the robot CoGiRo, a redundantly-actuated cable-suspended CDPR developed by Tecnalia and LIRMM [1]. The developed method is integrated within the open-source software Humanoid Path Planner (HPP) [2] which includes sampling-based planning algorithms like Probabilistic RoadMaps (PRM) and Rapidly-exploring Random Trees (RRT) [**?**], and different optimization methods. Those methods are already implemented and ready-to-use with few modifications needed for CDPRs, which is one of the main motivations for using HPP.

The issue of CDPR workspace analysis has been well studied, often with the assumption of a constant orientation of the platform and a simple straight line cable model [3], [4], [5], sometimes with more complex cable models taking into account elasticity or sagging [6]. The workspace can be defined by the equations of its boundaries [7], [8] or by sampling the global workspace and testing a large number of positions and orientations of the mobile platform [9]. The design of a cable-driven robot can be optimized so that the workspace is suited to a given task. In [10], the norm of the cable tension vector is minimized along a desired trajectory to select a cable configuration suited to the task. Concerning CDPR path planning, Tempel et al. [11] presented a very practical approach which allows artist to design trajectories of the mobile platform as Bézier curves, with an automatic path verification based on [12]. In this approach, a speficic tension distribution is tested. If this tension distribution is not feasible, the algorithm may fail to validate the path even though other feasible tension distributions may exist. The present work is closer to [13], where trajectories are

validated by expressing the tensions limits as algebraic inequalities and injecting the parametric trajectories into these inequalities, in order to obtain global conditions on the trajectory parameters. This method is applied to dynamic paths of a planar two-dimensional CDPR with a point-mass mobile platform. Merlet [14] showed that CDPR trajectories can be analysed to detect changes in the set of cables under tension, for example when a cable becomes slack along a trajectory. A discrete-time simulator is presented in [15] which checks trajectories for these changes. These methods focus on analyzing the evolution of a particular tension distribution along a path which is already known to be feasible, while the method proposed in the present paper aims at validating whether or not there exists feasible cable tensions along a path.

The contribution of this paper is a continuous cable tension validation method for constant-velocities trajectories integrated within the open-source software HPP. The method never fails to validate a path for which there exists at least one feasible cable tension distribution. Moreover, if a path is found non-valid, it means that there is absolutely no feasible tension distribution. The method proposed in this paper complements the continuous cable collision checking approach proposed in our previous work [16], aiming at a complete continuous path validation method for CDPRs. Moreover, its implementation in HPP makes it easy to use with other CDPR configurations and/or payloads, provided that the number of cables is at least equal to the number of mobile platform DOFs.

The general method presented in the present paper is inspired from [17], where Schwarzer et al. proposed to check collision of a multi-arm robot along a continuous path composed of linear interpolations in the joint space. The method is notably based on the computation of upper bounds on the relative velocities – linear and angular velocities – of each body in the reference frame of the other bodies. Then, given the distance – or a lower bound on the distance – between two bodies at a given parameter value, the method computes a time interval over which no collision can occur between the bodies. Our previous paper [16] extended this method to cable collisions (with the environment, with the mobile platform, and among them). In this paper, a similar approach is taken to validate the feasability of the cable tensions of a CDPR, by reasoning in the wrench space using the hyperplane shifting method [18].

The Cartesian space of the CDPR consists of seven values defining the pose of the mobile platform: three values for the translation and four values of a quaternion for the rotation. The method takes as input:

- A path of the CDPR mobile platform which is a linear interpolation in the Cartesian space, called in this paper a *straight path*.
- The weight-induced wrench applied on the platform along the path.

The method checks the validity of the path: it is not valid if there exists, along the path, at least one pose where the
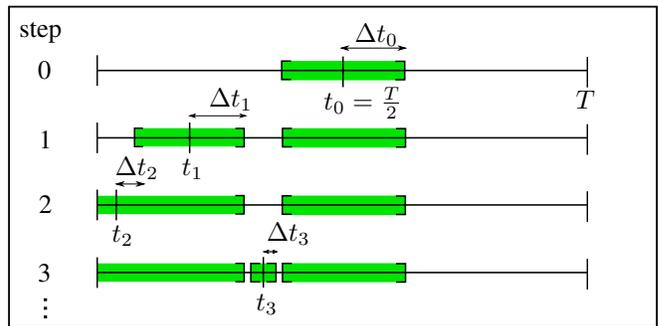

Fig. 2: Example of first four steps of the continuous validation of the path $[0; T]$. The validated parts of the interval are shown in green, the rest of the path remains to be checked.

mobile platform weight cannot be balanced with feasible cable tensions. A non-valid path is illustrated in Fig. 1.

In Section II, the general algorithm to continuously validate a path is briefly recalled. Section III then introduces useful notations and definitions, and details the calculations to find a valid interval around a given pose using the hyperplane shifting method. Implementations results in HPP with the CDPR CoGiRo are presented in Section IV and Section V concludes the paper.

II. CONTINUOUS VALIDATION ALGORITHM

We consider the general continuous validation algorithm presented in [16]. The algorithm can determine if a *straight path* is valid or not regarding a list of validation elements. A validation element can be a pair of bodies which must not collide for example, or in the case of the present paper, cable tensions which must be feasible. Each validation element has a method which takes as input a path and a joint configuration at a time parameter along this path, and returns a valid time interval around this parameter. The method tries and validates intervals at different time parameters on the path, until either a non-valid configuration is found, or all the path has been covered by the validated intervals.

Let us consider the *straight path* $[0; T]$. The algorithm starts by validating the path around $t_0 = \frac{T}{2}$. Either the pose at parameter t_0 is non-valid and the algorithm stops, or it is valid and the method finds a valid interval of length $2 * \Delta t_0$ around t_0. At this point, a portion $[\frac{T}{2} - \Delta t_0; \frac{T}{2} + \Delta t_0]$ has been validated and the intervals $[0; \frac{T}{2} - \Delta t_0]$ and $[\frac{T}{2} + \Delta t_0; T]$ remain to be validated. The algorithm proceeds by dichotomy: the middle of the first non-validated interval is taken as the next time parameter $t_1 = \frac{\frac{T}{2} - \Delta t_0}{2}$ and the algorithm checks the validity of an interval around t_1. When validated intervals overlap, they are merged. The algorithm iterates until a non-valid pose at a parameter t_i is found or the whole path is validated. The process is illustrated in Fig. 2, which shows the first four steps of the validation.

The next section details how the half-length of the valid interval around a time parameter is obtained for the validation of the feasibility of the cable tensions of a CDPR.
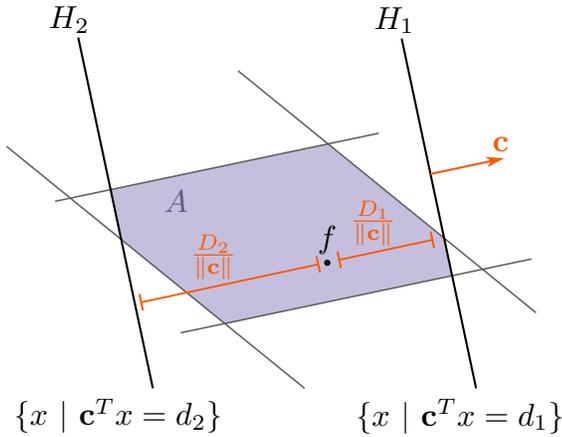
Fig. 3: A two-dimensional zonotope A with three pairs of parallel facet-defining hyperplanes.

Fig. 4: Notations for a CDPR.

III. EXTENTION OF THE HYPERPLANE SHIFTING METHOD

For a given pose and being given minimum and maximum allowed cable tensions, respectively $\tau_i^{\min}$ and $\tau_i^{\max}$ for cable i, the set of wrenches that a CDPR is able to generate, i.e. with cable tensions lying between minimum and maximum allowed tensions, is called the available wrench set (AWS), denoted A. The hyperplane shifting method [18] [19] allows the determination of the AWS. This wrench set is a zonotope, i.e. a particular type of convex polytope with pairs of parallel facets, as shown in Fig. 3. The hyperplane shifting method gives the equations of the facet-defining hyperplanes, which can be used to check if the required wrench set needed for a task is inside the zonotope and thereby feasible. Let us assume that the required wrench set is reduced to a single wrench $\mathbf{f}$. For a given pose, if the wrench $\mathbf{f}$ is inside the zonotope, then the pose is valid regarding the cable tensions. Along a path, the required wrench and the hyperplanes vary. To extend the static test to the continuous validation method, we define for each facet-defining hyperplane H a "distance" D whose positivity means that $\mathbf{f}$ lies in the feasible half-space delimited by H. An upper bound $V^{\max}$ on the time derivative of D is computed, valid along the whole path. These two values allow to compute the half-length Δt of a time interval where $\mathbf{f}$ is guaranteed to remain in the (feasible) half-space defined by the hyperplane H:

$$\Delta t = \frac{D}{V^{\max}} \quad (1)$$

By taking the minimum of these half-lengths for all the facet-defining hyperplanes, we obtain the half-length of an interval where the tensions in all cables are feasible. This paper deals with the case where the required wrench is the weight of the platform, which corresponds to the case of quasi-static motions of the CDPR.

A. Definitions and notations

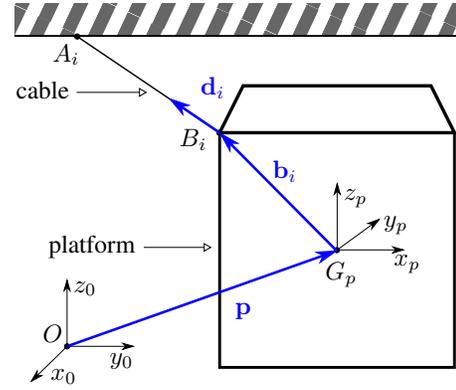Let us consider a CDPR composed of a mobile platform with n degrees of freedom, suspended on m cables, with $m \geq n$. In the rest of this paper, we consider the case $m = 8$ and $n = 6$ but the proposed method can be used for $n < 6$. A classic suspended CDPR is represented in Fig. 4. The cable elasticity and mass are neglected and all cables are assumed to be under tension, thus there is no sagging of the cables. It is also assumed that each cable exits the fixed structure at a fixed point A_i and is attached to the mobile platform at a point B_i, fixed in the platform frame. The constant vector $\mathbf{a}_i$ denotes the position of point A_i expressed in the global frame, while $\mathbf{b}_i$ denotes the position of B_i expressed in the platform frame. The position of the the mobile platform expressed in the global frame is denoted $\mathbf{p}$.

The total mass m_T — the mass of the platform and its optional payload — is fixed and constant along the path. Point G_p, the center of mass of the platform, is the origin of the platform frame and the reference point for moments. The external wrench $\mathbf{f}_g$ applied on the platform is the wrench induced by the platform and payload weights. It implies that $\mathbf{f}_g$ is constant in the fixed reference frame, with $\mathbf{f}_g = [0, 0, -m_T g, 0, 0, 0]^T$ where g is the gravity acceleration.

The wrench matrix of the robot is noted W, has size 6×8 and is given by the following expression:

$$W = \begin{bmatrix} \mathbf{d}_1 & \mathbf{d}_2 & \cdots & \mathbf{d}_8 \\ Q\mathbf{b}_1 \times \mathbf{d}_1 & Q\mathbf{b}_2 \times \mathbf{d}_2 & \cdots & Q\mathbf{b}_8 \times \mathbf{d}_8 \end{bmatrix} \quad (2)$$

where $\mathbf{d}_i$ is the unit vector directing cable i, and Q is the rotation matrix defining the orientation of the platform. The vector $\mathbf{w}_i$ denotes the i-th column of W: $\mathbf{w}_i = [\mathbf{d}_i^T \ (Q\mathbf{b}_i \times \mathbf{d}_i)^T]^T$.

The presented method is intended for use with path planning algorithms like RRT, which generate *straight paths* from random poses and uses the proposed method to validate them. Therefore, the mobile platform is considered to have a constant linear velocity $\mathbf{v}_p$ and constant angular velocity $\boldsymbol{\omega}_p$ along the input path, of norm respectively v_p and ω_p.

B. Facet-defining hyperplane of the AWS

The relationship between the wrench $\mathbf{f}$ applied by the cables on the platform and the cable tensions $\boldsymbol{\tau} = [\tau_1, ..., \tau_8]^T$

is:
$$\mathbf{f} = W\boldsymbol{\tau} \tag{3}$$

Along a quasi-static motion, the static equilibrium of the platform holds and $\mathbf{f} + \mathbf{f}_g = \mathbf{0}$. Let us consider, for a given pose x of the CDPR, the AWS $A = \{\mathbf{f} \mid \mathbf{f} = W\boldsymbol{\tau}, \tau_i^{\min} \le \tau_i \le \tau_i^{\max}, \forall i \in [1,8]\}$. Since A is the image of a box by a linear application, A is a zonotope, with pairs of parallel facet-defining hyperplanes. As shown in [19], each such pair of hyperplanes H_1 and H_2 corresponds to a combination of $n-1 = 5$ linearly independent columns w_i of the wrench matrix W defined in (2). Let I^0 be the set of the indices of these columns and W_0 be the corresponding submatrix of W. According to [19], the two hyperplanes have the following equations:

$$H_1 = \{\mathbf{f} \mid \mathbf{c}^T\mathbf{f} = d_1\} \quad H_2 = \{\mathbf{f} \mid \mathbf{c}^T\mathbf{f} = d_2\} \tag{4}$$

where $\mathbf{c}$ is any non-zero vector orthogonal to the set $(\mathbf{w}_i)_{i\in I^0}$, and d_1 and d_2 are given by:

$$d_1 = \sum_{I^+} \tau_i^{\max}\mathbf{c}^T\mathbf{w}_i + \sum_{I^-} \tau_i^{\min}\mathbf{c}^T\mathbf{w}_i \tag{5}$$

$$d_2 = -\sum_{I^-} \tau_i^{\max}\mathbf{c}^T\mathbf{w}_i - \sum_{I^+} \tau_i^{\min}\mathbf{c}^T\mathbf{w}_i \tag{6}$$

with:
- $I^+ = \{i \mid \mathbf{c}^T\mathbf{w}_i > 0\}$
- $I^- = \{i \mid \mathbf{c}^T\mathbf{w}_i < 0\}$

The vector $\mathbf{c}$ can be chosen by means of any routine that determines the nullspace of a matrix. The vectors $\mathbf{w}_i$ and $\mathbf{c}$ and the values d_1 and d_2 are continuous along a path. The value $\mathbf{c}^T\mathbf{w}_i$ for a cable i can change sign along a path, which means that the sets I^+ and I^- can change.

The wrench $\mathbf{f}$ is in the AWS A if and only if for all hyperplanes $H_1 = \{\mathbf{f} \mid \mathbf{c}^T\mathbf{f} = d_1\}$ and $H_2 = \{\mathbf{f} \mid \mathbf{c}^T\mathbf{f} = d_2\}$, we have:

$$\mathbf{c}^T\mathbf{f} - d_1 < 0 \quad \text{and} \quad -\mathbf{c}^T\mathbf{f} - d_2 < 0 \tag{7}$$

For each pair of hyperplanes, the goal is to compute, for a given pose of the robot along a path, an interval around that pose over which the conditions of (7) are respected.

For the hyperplane shifting method to work, the matrix W must have at least n linearly independent columns. For Cogiro, with $n = 6$ and $m = 8$, there can be up to 3 linearly dependent columns. If there are more than 3, the robot is in a singularity and cannot control all its DOFs.

C. "Distance" D between f and a hyperplane H

For a given pose of the CDPR and a pair of facet-defining hyperplanes H_1 and H_2 defined in (4), we define the values D_1 and D_2:

$$D_1 = -\mathbf{c}^T\mathbf{f} + d_1 \quad D_2 = \mathbf{c}^T\mathbf{f} + d_2 \tag{8}$$

The pose is valid if for all pairs of hyperplanes H_1 and H_2, D_1 and D_2 are positive. It should be noted that D_1 and D_2 depend on the vector $\mathbf{c}$, and are not the 6-dimensional euclidian distances between $\mathbf{f}$ and the hyperplanes H_1 and H_2, which are respectively $\frac{D_1}{\|\mathbf{c}\|}$ and $\frac{D_2}{\|\mathbf{c}\|}$.

Without loss of generality, let us assume that the index set I^0 is equal to $\{1,2,3,4,5\}$. Let us consider $V = W_0^T$, so that the rows of V are the transposes of the 5 column vectors $(\mathbf{w}_i)_{i\in I^0}$. Since the set $(\mathbf{w}_i)_{i\in I^0}$ is linearly independent, V is a matrix of size 5×6 with full rank 5. Let $\mathbf{v}_i$ be the columns of V. The nullspace of V has dimension 1 and (9) gives a determinant-based expression of a vector $\mathbf{c}$ spanning this nullspace, e.g. $\ker(V) = \mathrm{span}(\mathbf{c})$ [20]. Such a vector $\mathbf{c}$ satisfies $V\mathbf{c} = \mathbf{0}$, which means that $\mathbf{c}$ is orthogonal to the set $(\mathbf{w}_i)_{i\in I^0}$.

$$\mathbf{c} = \begin{bmatrix} \det([\mathbf{v}_2\mathbf{v}_3\mathbf{v}_4\mathbf{v}_5\mathbf{v}_6]) \\ - \quad \det([\mathbf{v}_1\mathbf{v}_3\mathbf{v}_4\mathbf{v}_5\mathbf{v}_6]) \\ \det([\mathbf{v}_1\mathbf{v}_2\mathbf{v}_4\mathbf{v}_5\mathbf{v}_6]) \\ - \quad \det([\mathbf{v}_1\mathbf{v}_2\mathbf{v}_3\mathbf{v}_5\mathbf{v}_6]) \\ \det([\mathbf{v}_1\mathbf{v}_2\mathbf{v}_3\mathbf{v}_4\mathbf{v}_6]) \\ - \quad \det([\mathbf{v}_1\mathbf{v}_2\mathbf{v}_3\mathbf{v}_4\mathbf{v}_5]) \end{bmatrix} \tag{9}$$

The ith element of $\mathbf{c}$, $1 \le i \le 6$, is equal to $(-1)^{i+1}\det_i$ where $\det_i$ is the determinant of the 5×5 square matrix obtained from the matrix V by deleting its ith column. Since the matrix V has full rank, $\mathbf{c}$ defined in (9) is a non-zero vector. Using this expression of $\mathbf{c}$, we can easily compute D_1 and D_2 from (5), (6) and (8). It should be noted that $\mathbf{c}$ is deliberately not normalized in order to facilitate the computation of the upper bound on the derivative of $\mathbf{c}$ in the next paragraph.

D. Upper bound V^{max} on the derivative of D for a hyperplane H

For a given pose and a pair of facet-defining hyperplanes H_1 and H_2, the time derivatives of the values D_1 and D_2 are obtained by differentiating (8):

$$\dot{D}_1 = -\dot{\mathbf{c}}^T\mathbf{f} - \mathbf{c}^T\dot{\mathbf{f}} + \dot{d}_1 \quad \dot{D}_2 = \dot{\mathbf{c}}^T\mathbf{f} + \mathbf{c}^T\dot{\mathbf{f}} + \dot{d}_2 \tag{10}$$

As stated in paragraph III-A, we assume the platform wrench $\mathbf{f}$ is constant and known, which gives: $\dot{\mathbf{f}} = \mathbf{0}$. Using the triangle inequality, we obtain:

$$|\dot{D}_1| \le \|\dot{\mathbf{c}}\|\|\mathbf{f}\| + |\dot{d}_1| \quad |\dot{D}_2| \le \|\dot{\mathbf{c}}\|\|\mathbf{f}\| + |\dot{d}_2| \tag{11}$$

By differentiating (5) and (6):

$$\dot{d}_1 = \sum_{I^+} \tau_i^{\max}(\dot{\mathbf{c}}^T\mathbf{w}_i + \mathbf{c}^T\dot{\mathbf{w}}_i) + \sum_{I^-} \tau_i^{\min}(\dot{\mathbf{c}}^T\mathbf{w}_i + \mathbf{c}^T\dot{\mathbf{w}}_i) \tag{12}$$

$$\dot{d}_2 = -\sum_{I^-} \tau_i^{\max}(\dot{\mathbf{c}}^T\mathbf{w}_i + \mathbf{c}^T\dot{\mathbf{w}}_i) - \sum_{I^+} \tau_i^{\min}(\dot{\mathbf{c}}^T\mathbf{w}_i + \mathbf{c}^T\dot{\mathbf{w}}_i) \tag{13}$$

Since we consider a fixed combination of 5 vectors $(\mathbf{w}_i)_{i\in I_0}$ for which $\mathbf{c}^T\mathbf{w}_i = 0$, we know that $I^+ \cup I^-$ is constant. For $i \in I^+ \cup I^-$, the absolute values of the corresponding terms of $\dot{d}_1$ and $\dot{d}_2$ are either $\tau_i^{\max}|\dot{\mathbf{c}}^T\mathbf{w}_i + \mathbf{c}^T\dot{\mathbf{w}}_i|$ or $\tau_i^{\min}|\dot{\mathbf{c}}^T\mathbf{w}_i + \mathbf{c}^T\dot{\mathbf{w}}_i|$. Moreover, $\tau_i^{\min} \le \tau_i^{\max}$ gives:

$$|\dot{d}_{\delta\in\{1,2\}}| \le \sum_{i\in I^+\cup I^-} \tau_i^{\max}|\dot{\mathbf{c}}^T\mathbf{w}_i + \mathbf{c}^T\dot{\mathbf{w}}_i| \tag{14}$$

We consider the following upper bounds:
- $w_i^{\max}$ is an upper bound on $\|\mathbf{w}_i\|$.
- $\dot{w}_i^{\max}$ is an upper bound on $\|\dot{\mathbf{w}}_i\|$.
- $c^{\max}$ is an upper bound on $\|\mathbf{c}\|$.

- $\dot{c}^{\max}$ is an upper bound on $\|\dot{\mathbf{c}}\|$.

An upper bound $V^{\max}$ on the norms of $\dot{D}_1$ and $\dot{D}_2$ is obtained using these bounds and the fact that $\|\mathbf{f}\| = \|-\mathbf{f}_g\| = m_T g$:

$$V^{\max} = \dot{c}^{\max} m_T g \tag{15}$$
$$+ \sum_{i \in I^+ \cup I^-} \tau_i^{\max} (\dot{c}^{\max} w_i^{\max} + c^{\max} \dot{w}_i^{\max})$$

We now have to find values for the upper bounds defined above.

1) Upper bound w_i^{max} on $\|\mathbf{w}_i\|$: A vector $\mathbf{w}_i$ is a column vector containing the unit forces and torques applied by cable i to the platform and has the following expression:

$$\mathbf{w}_i = \begin{bmatrix} \mathbf{d}_i \\ Q\mathbf{b}_i \times \mathbf{d}_i \end{bmatrix} \tag{16}$$

By definition, $\mathbf{d}_i$ is a unit vector and Q is a rotation matrix and thus have norm 1. We can write $\|\mathbf{w}_i\|^2 = \|\mathbf{d}_i\|^2 + \|Q\mathbf{b}_i \times \mathbf{d}_i\|^2$, which gives an upper bound $w_i^{\max}$ on the norm of $\mathbf{w}_i$:

$$\|\mathbf{w}_i\| \le \sqrt{1 + b_i^2} = w_i^{\max} \tag{17}$$

where b_i is the constant norm of vector $\mathbf{b}_i$, fixed by the CDPR design.

2) Upper bound $\dot{w}_i^{max}$ on $\|\dot{\mathbf{w}}_i\|$: Differentiating the expression of $\mathbf{w}_i$ in (16) gives:

$$\dot{\mathbf{w}}_i = \begin{bmatrix} \dot{\mathbf{d}}_i \\ [\boldsymbol{\omega}_p]_\times Q\mathbf{b}_i \times \mathbf{d}_i + Q\mathbf{b}_i \times \dot{\mathbf{d}}_i \end{bmatrix} \tag{18}$$

where $[\boldsymbol{\omega}_p]_\times$ is the cross product matrix associated to the platform angular velocity vector $\boldsymbol{\omega}_p$. We need to find upper bounds on the norms of the two vectors composing $\dot{\mathbf{w}}_i$. For $\dot{\mathbf{d}}_i$, we have by definition of $\mathbf{d}_i$:

$$\mathbf{d}_i = \frac{\overrightarrow{B_i A_i}}{l_i} \tag{19}$$

with $\|\overrightarrow{B_i A_i}\| = l_i$. The formula for the derivative of a unit vector $\mathbf{u} = \frac{\mathbf{v}}{\|\mathbf{v}\|}$, $\dot{\mathbf{u}} = \frac{1}{\|\mathbf{v}\|}(I - \mathbf{u}\mathbf{u}^T)\dot{\mathbf{v}}$, applied to $\mathbf{d}_i$, gives:

$$\dot{\mathbf{d}}_i = \frac{1}{l_i}(I - \mathbf{d}_i \mathbf{d}_i^T)\frac{d\overrightarrow{B_i A_i}}{dt} \tag{20}$$

Since $\overrightarrow{B_i A_i} = \mathbf{a}_i - Q\mathbf{b}_i - \mathbf{p}$, we have $\frac{d\overrightarrow{B_i A_i}}{dt} = -\dot{Q}\mathbf{b}_i - \mathbf{v}_p$ because $\mathbf{a}_i$ and $\mathbf{b}_i$ are constant vectors. This gives $\|\frac{d\overrightarrow{B_i A_i}}{dt}\| \le \omega_p b_i + v_p$. We consider a constant non-negative minimum possible length for each cable of the CDPR: $L_i^{\min} \le l_i$.

The vector $\mathbf{d}_i$ has a norm of 1 and $\|\mathbf{d}_i \mathbf{d}_i^T\| \le \|\mathbf{d}_i\|^2$, therefore $\|\mathbf{d}_i \mathbf{d}_i^T\| \le 1$. The following expression for the upper bound $\dot{d}_i^{\max}$ on $\|\dot{\mathbf{d}}_i\|$ is obtained:

$$\|\dot{\mathbf{d}}_i\| \le \frac{2}{L_i^{\min}}(\omega_p b_i + v_p) = \dot{d}_i^{\max} \tag{21}$$

For the norm of the derivative of the torque in (18), the following inequality can be written:

$$\|[\boldsymbol{\omega}_p]_\times Q\mathbf{b}_i \times \mathbf{d}_i + Q\mathbf{b}_i \times \dot{\mathbf{d}}_i\| \le \|[\boldsymbol{\omega}_p]_\times\|\|Q\|\|\mathbf{b}_i\|\|\mathbf{d}_i\| + \|Q\|\|\mathbf{b}_i\|\|\dot{\mathbf{d}}_i\| \tag{22}$$

By definition $\|\mathbf{d}_i\| = 1$ and $\|Q\| = 1$, and the norm of $\boldsymbol{\omega}_p$ is constant along the path by hypothesis. Inequality (22) can be rewritten as:

$$\|[\boldsymbol{\omega}_p]_\times Q\mathbf{b}_i \times \mathbf{d}_i + Q\mathbf{b}_i \times \dot{\mathbf{d}}_i\| \le \omega_p b_i + b_i \dot{d}_i^{\max} \tag{23}$$

The norm of $\dot{\mathbf{w}}_i$ can be expressed using (18): $\|\dot{\mathbf{w}}_i\| = \sqrt{\|\dot{\mathbf{d}}_i\|^2 + \|[\boldsymbol{\omega}_p]_\times Q\mathbf{b}_i \times \mathbf{d}_i + Q\mathbf{b}_i \times \dot{\mathbf{d}}_i\|^2}$, which provides the following inequality defining the upper bound $\dot{w}_i^{\max}$ on the norm of $\mathbf{w}_i$:

$$\|\dot{\mathbf{w}}_i\| \le \sqrt{\dot{d}_i^{\max} + b_i^2(\omega_p + \dot{d}_i^{\max})^2} = \dot{w}_i^{\max} \tag{24}$$

3) Upper bound c^{max} on $\|\mathbf{c}\|$: The vector $\mathbf{c}$ is defined in (9). Let us consider the square matrix V_i whose columns are the vectors $\mathbf{v}_j$ with $j \ne i$. For example, for V_1, we have:

$$\det([\mathbf{v}_2\mathbf{v}_3\mathbf{v}_4\mathbf{v}_5\mathbf{v}_6]) = \det(V_1) \tag{25}$$
$$= \det(V_1^T) = \det([\mathbf{u}_1^1\mathbf{u}_2^1\mathbf{u}_3^1\mathbf{u}_4^1\mathbf{u}_5^1]) \tag{26}$$

where the $\mathbf{u}_k^i$, $1 \le k \le 6$ are the columns of V_i^T (i.e. the rows of V_i). The vector $\mathbf{u}_k^i$ is equal to the vector $\mathbf{w}_k$ with the i-th coefficient missing. Then:

$$\mathbf{c} = \begin{bmatrix} \det([\mathbf{u}_1^1\mathbf{u}_2^1\mathbf{u}_3^1\mathbf{u}_4^1\mathbf{u}_5^1]) \\ -\quad \det([\mathbf{u}_1^2\mathbf{u}_2^2\mathbf{u}_3^2\mathbf{u}_4^2\mathbf{u}_5^2]) \\ \det([\mathbf{u}_1^3\mathbf{u}_2^3\mathbf{u}_3^3\mathbf{u}_4^3\mathbf{u}_5^3]) \\ -\quad \det([\mathbf{u}_1^4\mathbf{u}_2^4\mathbf{u}_3^4\mathbf{u}_4^4\mathbf{u}_5^4]) \\ \det([\mathbf{u}_1^5\mathbf{u}_2^5\mathbf{u}_3^5\mathbf{u}_4^5\mathbf{u}_5^5]) \\ -\quad \det([\mathbf{u}_1^6\mathbf{u}_2^6\mathbf{u}_3^6\mathbf{u}_4^6\mathbf{u}_5^6]) \end{bmatrix} \tag{27}$$

Knowing that $\mathbf{u}_k^i$ is a vector of dimension 5 and $\mathbf{w}_k$ is a vector of dimension 6 with the same elements as $\mathbf{u}_k$ and one more element, we can write $\|\mathbf{u}_k^i\| \le \|\mathbf{w}_k\|$ and $\|\dot{\mathbf{u}}_k^i\| \le \|\dot{\mathbf{w}}_k\|$.

Using Hadamard's inequality, which states that the norm of a determinant of a matrix is not bigger than the product of the norms of its column vectors, we get:

$$\|\mathbf{c}\| \le \sqrt{\sum_{i=1}^{i=6}\left(\prod_{k=1}^{k=5} w_k^{\max}\right)^2} \tag{28}$$

Hence, the following expression for the upper bound $c^{\max}$ on the norm of $\dot{\mathbf{c}}$ is obtained:

$$\|\mathbf{c}\| \le c^{\max} = \sqrt{6} \prod_{k=1}^{k=5} w_k^{\max} \tag{29}$$

4) Upper bound $\dot{c}^{max}$ on $\|\dot{\mathbf{c}}\|$: The expression of $\mathbf{c}$ is given in (27). The formula of the derivative of the determinant of a matrix and the Laplace expansion provide an expression of the i-th coefficient of $\dot{\mathbf{c}}$:

$$\dot{c}_i = (-1)^{i+1}\sum_{j=1}^{j=5}\det_u^{i,j} \tag{30}$$

with $\det_u^{i,j} = \det(\mathbf{u}_1^i, ..., \dot{\mathbf{u}}_j^i, ..., \mathbf{u}_5^i)$. Using again Hadamard's inequality, $\|\det_u^{i,j}\|$ can be bounded:

$$|\det_u^{i,j}| \le \|\dot{\mathbf{u}}_j^i\| \prod_{k-1,\ne j}^{k=n-1} \|\mathbf{u}_k^i\| \tag{31}$$

Using again the inequalities $\|\mathbf{u}_k^i\| \leq \|\mathbf{w}_k\|$ and $\|\dot{\mathbf{u}}_k^i\| \leq \|\dot{\mathbf{w}}_k^i\|$, the following bound on $\|\dot{\mathbf{c}}\|$ is obtained:

$$\|\dot{\mathbf{c}}\| \leq \sqrt{\sum_{i=1}^{i=6} \left(\sum_{j=1}^{j=5} \left(\|\dot{\mathbf{w}}_j\| \prod_{k=1,k\neq j}^{k=5} \|\mathbf{w}_k\| \right) \right)^2} \quad (32)$$

Thus $\|\dot{\mathbf{c}}\| \leq \dot{c}^{\max}$ can be written as:

$$\dot{c}^{\max} = \sqrt{6} \sum_{j=1}^{j=5} \left(\dot{w}_j^{\max} \prod_{k=1,k\neq j}^{k=5} w_k^{\max} \right) \quad (33)$$

Using (17), (24), (29) and (33), $V^{\max}$ defined in (15) can be computed. With D_1 and D_2 defined in (8), the values $\Delta t_1 = \frac{D_1}{V^{\max}}$ and $\Delta t_2 = \frac{D_2}{V^{\max}}$ are obtained. Repeating these computations for every pair of facet-defining hyperplanes and taking the lowest value Δt gives the half-length of a valid interval.

IV. IMPLEMENTATION RESULTS

The continuous validation algorithm presented in Section II and the computations presented in Section III are implemented in the software Humanoid Path Planner (HPP). To find a valid trajectory between two given poses of the robot, several path planning methods such as RRT [21] can be used. These methods then call the validation method proposed in this paper to validate smaller paths, and concatenate them to obtain a global valid path.

The length of the path is expressed in a non-homogeneous unit corresponding to the variation in position and orientation of the mobile platform. The CDPR static equilibirum workspace with varying orientation is not convex: for a *straight path* between two valid poses, a non-valid interval may exist along the path. The proposed method is able to detect any such non-valid intervals along a path, no matter how short the interval is.

Random *straight paths* were generated by shooting random valid poses of the mobile platform in the workspace, and checked using the continuous methods. Table I shows computation times for the continuous validation of $N = 1000$ random paths of same length 1. If the pose at the middle of a path is non-valid, then the continuous method finishes very quickly, as shows the minimum computation time. Paths that take the longest to validate are paths along which the mobile platform approaches (or crosses) the border of the valid workspace. In these cases, the required wrench $\mathbf{f}$ gets closer and closer to the boundaries of the AWS, and therefore the algorithm validates smaller and smaller intervals, which results in a longer computation time.

Fig. 1 shows different poses of the CDPR CoGiRo along an example path. The poses at the extremities of the path are valid. This example path has a total length of $T = 5$. Using the continuous validation method, it is shown that the path is not valid and that there exists a non-valid interval of length $d = 0.09$. If the same path is checked using a discretized validation with a time step larger than d, for example a default value of 0.1, it may be wrongly validated

	nb of paths	computation times (s)		
		min	mean	max
Valid paths	994	1.21	4.51	29.52
Non-valid paths	6	4.31e-04	2.45	14.68
All paths	1000		4.49	

TABLE I: Results for the continuous tension validation of 1000 *straight paths* of length 1.

and constitute a false positive. The continous method never produces such false positives.

It should be noted that by choosing a small enough time step, a discretized validation method can always detect the non-valid intervals. This is however at the cost of larger computation times since, for discretized validation, the latter is inversely proportional to the time step. Moreover, the time step is often fixed to a default value and it is not easy to tune it and obtain a compromise between speed and correctness.

The numerical results show that the upper bound $V^{\max}$ on the derivative of D obtained in (15) is not a tight bound: there is a factor of 10^3 between $V^{\max}$ and the observed values of $\dot{D}$. For CoGiRo, paths can have length up to $T = 15$. Tested on random paths of random lengths, the continuous validation takes up to two minutes to validate a random path, while a discretized method with a default time step of 0.01 takes up to 0.5 seconds. Although the difference in speed is important, the continuous method offers garantees that are crucial in an industrial setting and that a discretized method does not offer.

Due to the dichotomic nature of the validation algorithm, when the robot comes close to a non-valid configuration whithout reaching it, the validation takes longer because the validated intervals get smaller and smaller. This means that simpler path are in general faster to validate.

For the CDPR CoGiRo, the majority of the tested paths that were found non-valid regarding cable tensions are also non-valid regarding cable collisions. This shows that the workspace of CoGiRo is limited by the cable collisions more than by the cable tension limits. The path illustrated in Fig. 1 was tested using the proposed continuous cable tension validation method combined with the continuous cable collision validation method proposed previously in [16].

V. CONCLUSIONS

The presented continuous tension validation method is able to guarantee that the cable tensions are feasible along a path. While a discretized method may produce false positives, the presented method never does. Used with the continuous collision checking method presented in [16], it constitutes a complete continuous validation method for CDPRs integrated within the open source software HPP and can be used for automatic path planning.

The proposed method may be especially useful for CDPRs whose design could not be optimized regarding the cable ten-

sions. For robots such as CoGiRo, which have been designed to optimize wrench capabilities across their workspace, the method is useful when validating movements in the limits of the workspace.

Future works will focus on improving the upper bound $V^{\max}$ used in the proposed continuous validation method, and thereby improving the computation time, and on extending this method to paths with a varying external wrench, for example by considering a robotic arm mounted on the platform of a CDPR. Such an hybrid CDPR should yield more cases where a *straight path* between two valid poses contains a non-valid interval, for which the proposed method proves useful. It is also possible to modify the algorithm to consider a larger required wrench set not limited to a single point, in order to take into account the uncertainty of the pose of the robot or the dynamics of the movement.

REFERENCES

[1] M. Gouttefarde, J.-F. Collard, N. Riehl, and C. Baradat, "Geometry Selection of a Redundantly Actuated Cable-Suspended Parallel Robot," *IEEE Transactions on Robotics*, vol. 31, no. 2, pp. 501–510, Apr. 2015. [Online]. Available: https://hal-lirmm.ccsd.cnrs.fr/lirmm-01221397

[2] J. Mirabel, S. Tonneau, P. Fernbach, A.-K. Seppälä, M. Campana, N. Mansard, and F. Lamiraux, "HPP: a new software for constrained motion planning," in *IEEE/RSJ Intelligent Robots and Systems*, October 2016.

[3] G. Barrette and C. Gosselin, "Determination of the dynamic workspace of cable-driven planar parallel mechanisms," *Journal of Mechanical Design*, vol. 127, pp. 242–248, 03 2005.

[4] A. Berti, J.-P. Merlet, and M. Carricato, "Workspace analysis of redundant cable-suspended parallel robots," in *Cable-Driven Parallel Robots*, A. Pott and T. Bruckmann, Eds. Cham: Springer International Publishing, 2015, pp. 41–53.

[5] G. Abbasnejad, J. Eden, and D. Lau, "Generalized ray-based lattice generation and graph representation of wrench-closure workspace for arbitrary cable-driven robots," *IEEE Transactions on Robotics*, vol. 35, no. 1, p. 147161, 2019.

[6] J. Merlet, "On the workspace of suspended cable-driven parallel robots," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, May 2016, pp. 841–846.

[7] E. Stump and V. Kumar, "Workspaces of cable-actuated parallel manipulators," *Journal of Mechanical Design*, vol. 128, pp. 159–167, 01 2006.

[8] M. Gouttefarde and C. M. Gosselin, "Analysis of the wrench-closure workspace of planar parallel cable-driven mechanisms," *IEEE Transactions on Robotics*, vol. 22, no. 3, pp. 434–445, June 2006.

[9] X. Diao and O. Ma, "Workspace analysis of a 6-dof cable robot for hardware-in-the-loop dynamic simulation," in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Oct 2006, pp. 4103–4108.

[10] J. T. Bryson, X. Jin, and S. K. Agrawal, "Optimal design of cable-driven manipulators using particle swarm optimization," *ASME Journal of Mechanisms and Robotics*, vol. 8, no. 4, 2016.

[11] P. Tempel, F. Schnelle, A. Pott, and P. Eberhard, "Design and programming for cable-driven parallel robots in the german pavilion at the expo 2015," *Machines*, vol. 3, pp. 223–241, 08 2015.

[12] A. Pott, T. Bruckmann, and L. Mikelsons, "Closed-form force distribution for parallel wire robots," in *Computational Kinematics*, A. Kecskeméthy and A. Müller, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 25–34.

[13] C. Gosselin, Ping Ren, and S. Foucault, "Dynamic trajectory planning of a two-dof cable-suspended parallel robot," in *2012 IEEE International Conference on Robotics and Automation*, May 2012, pp. 1476–1481.

[14] J.-P. Merlet, "Checking the cable configuration of cable-driven parallel robots on a trajectory," 05 2014, pp. 1586–1591.

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

[16] D. Bury, J.-B. Izard, M. Gouttefarde, and F. Lamiraux, "Continuous collision detection for a robotic arm mounted on a cable-driven parallel robot," in *IEEE/RSJ Intelligent Robots and Systems (IROS)*, Macau, China, 2019.

[17] F. Schwarzer, M. Saha, and J.-C. Latombe, "Exact collision checking of robot paths," in *Algorithmic Foundations of Robotics V, STAR 7*, J.-D. B. et al., Ed. Springer, 2004, pp. pp 25–41.

[18] S. Bouchard, C. Ement, C. Gosselin, and B. Moore, "On the ability of a cable-driven robot to generate a prescribed set of wrenches," *Journal of Mechanisms and Robotics*, vol. 2, 02 2010.

[19] M. Gouttefarde and S. Krut, "Characterization of parallel manipulator available wrench set facets," in *Advances in Robot Kinematics: Motion in Man and Machine*, J. Lenarcic and M. M. Stanisic, Eds. Dordrecht: Springer Netherlands, 2010, pp. 475–482.

[20] M. Gouttefarde and C. M. Gosselin, "On the Properties and the Determination of the Wrench-Closure Workspace of Planar Parallel Cable-Driven Mechanisms," ser. International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, vol. Volume 2: 28th Biennial Mechanisms and Robotics Conference, Parts A and B, 06 2008, pp. 337–346.

[21] S. M. Lavalle, "Rapidly-exploring random trees: A new tool for path planning," Tech. Rep., 1998.