Determination of the Wrench-Closure Translational Workspace in Closed-Form for Cable-Driven Parallel Robots

Andreas Pott[1,2] and Werner Kraus[2]

***Abstract*— Workspace determination for robots is an important step in analysis and synthesis. A couple of methods for computing the wrench-closure workspace of cable-driven parallel robots were reported in the literature but all methods tend to be time consuming. In this paper, a new algorithm is presented that exploits different techniques to speed up the computation. Pre-computation is largely exploited and benefit are gained both from considerations in computer algebra and efficient numerical routines. Results from the computation of the translational (sometimes also called constant orientation) wrench-closure workspace are presented and performance values are provided. To the best of the authors' knowledge, the method proposed in this paper is superior in terms of computational time to any other approach for workspace computation.**

I. INTRODUCTION

Workspace evaluation for cable-driven parallel robots is a challenging task required amongst other for robot analysis and parameter synthesis. Early studies of suspended cable robots including workspace aspects were presented by Albus [1]. Verhoeven [16] shows that the wrench-closure workspace of completely and redundantly constrained cable robots is in general bounded by polynomial surfaces and also provides an explicit formula to compute the polynomials. Gouttefarde [4], [6] shows that the wrench-closure workspace of planar robots consists of conic sections and elaborates techniques to determine the boundaries of that workspace. Later, a technique was proposed to compute cross sections of the wrench-closure workspace of cable robots and it was showed that the translational workspace is bounded by cubic surfaces [7]. Gouttefarde [5] provides some theorems to characterize the boundary of the wrench-closure workspace for six d.o.f. robots with seven cables. Using other arguments, the results from Gouttefarde were generalized to spatial robots by Stump [15]. Hadian [8] studies the wrench-feasible workspace of a specific 6-6 suspended cable robot and derives explicit formulas for cross section of the translational workspace. Azizian [3], [2] determines the boundaries of the wrench-feasible workspace for planar robots. Hassan [9] presents an analytical expression for the wrench-closure workspace for the example of a storage receival machine. Therefore, the authors exploit the symmetric geometry of the

*The work was partly supported by the German Research Foundation (DFG) within the Cluster of Excellence in Simulation Technology (EXC 310/1) at the University of Stuttgart.

[1]Andreas Pott is with Institute for Control Engineering of Machine Tools and Manufacturing Units, University of Stuttgart, Seidenstr. 36, 70174 Stuttgart, Germany

[2]Andreas Pott and Werner Kraus are with the Department of Robot and Assistive Systems, Fraunhofer IPA, Nobelstr. 12, 70569 Stuttgart, Germany andreas.pott@ipa.fraunhofer.de

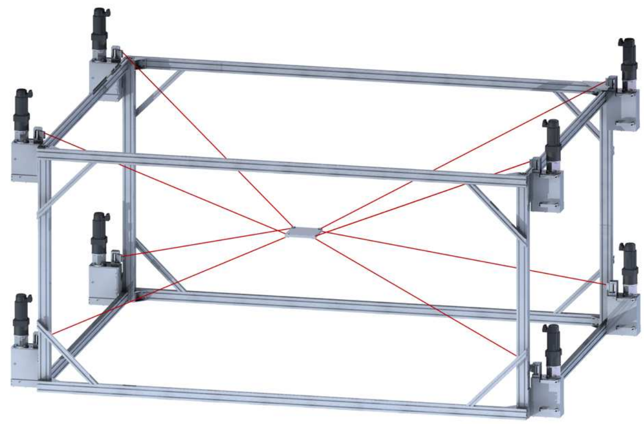

Fig. 1. CAD draft of the spatial cable robot IPAnema with eight cables and six d.o.f.

eight cables robot with six d.o.f. by essentially considering a cross section of the workspace to simplify the analysis of statics to an equivalent of a planar robot. Then, the separating hyperplane approach is used to compute the workspace. Recently, an algebraic form of the boundary of the wrench-closure workspace was determined from sub-determinants of the structure matrix using computer algebra by Sheng [14].

In this paper, we extend the latter approach [14] to compute the wrench-closure workspace based on the algebraic expressions. However, the approach presented here is based on a mixture of considerations on the algebraic structure where a numerical scheme is proposed that exploits the structure without using computer algebra to manipulate the equations. The proposed method employs relatively simple mathematical tooling. In contrast to Verhoeven's conjecture [16] that the closed-form of the wrench-closure workspace is practically useless due to is excessive length, we show in this contribution that firstly the algebraic form can be broken down to a handy data model and using a numerical procedure, this workspace can be computed extremely fast. Thus, applications in real-time systems are basically possible. Thus, the key advantage of the scheme presented here is its outstanding computation time to determine the shape, size, and volume of the wrench-closure translational workspace of robots with arbitrary geometry.

The rest of the paper is structured as follows. Sec. II recalls the relevant background used in this paper. In Sec. III, the idea and the algorithm for computing the analytic workspace model are derived. Then, an algorithm is proposed to generate a standard workspace representation from the analytic

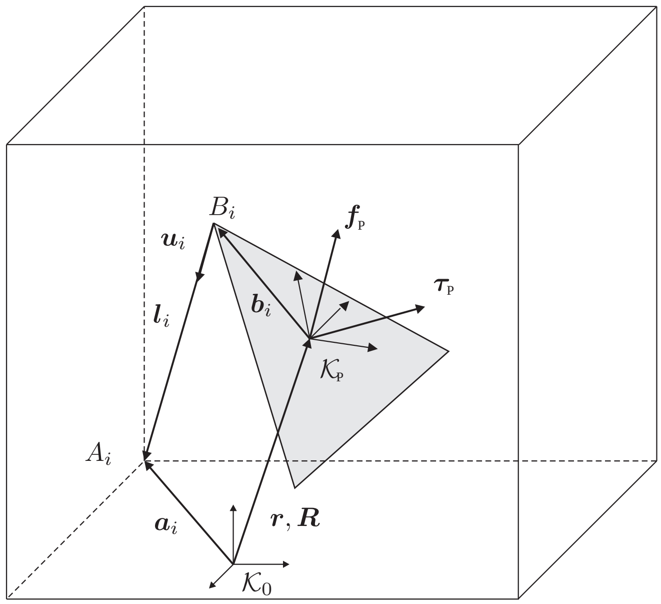

Fig. 2. Vector loop for geometry and kinematics of a general spatial cable robot

model (Sec. IV) and results are presented (Sec. V). The paper closes with some conclusions and future work.

II. KINEMATIC AND STATIC FUNDAMENTALS

A. Kinematic analysis

For better reference, the kinematics of cable robots is briefly reviewed. Fig. 2 shows the kinematic structure of a spatial cable robot, where the vectors $\boldsymbol{a}_i$ denote the proximal attachment points on the frame, the vectors $\boldsymbol{b}_i$ are the relative positions of the distal attachment points on the movable platform decomposed in the moving frame of the mobile platform, and l_i denote the length of the cables. Let m be the number of cables. Applying a vector loop, the closure-constraint reads

$$\boldsymbol{l}_i = \boldsymbol{a}_i - \boldsymbol{r} - \boldsymbol{R}\boldsymbol{b}_i \quad \text{for} \quad i = 1, \dots, m \quad , \tag{1}$$

where the vector $\boldsymbol{r}$ is the Cartesian position of the platform and the rotation matrix $\boldsymbol{R}$ represents the orientation of the platform. The unit vector along the cable becomes

$$\boldsymbol{u}_i = \frac{\boldsymbol{l}_i}{l_i} \quad \text{with} \quad l_i = |\boldsymbol{l}_i| \quad . \tag{2}$$

For force and torque equilibrium, it holds true [11], [16]

$$\underbrace{\begin{bmatrix} \boldsymbol{u}_1 & \dots & \boldsymbol{u}_m \\ \boldsymbol{R}\boldsymbol{b}_1 \times \boldsymbol{u}_1 & \dots & \boldsymbol{R}\boldsymbol{b}_m \times \boldsymbol{u}_m \end{bmatrix}}_{\boldsymbol{A}^{\mathrm{T}}(\boldsymbol{r},\boldsymbol{R})} \underbrace{\begin{bmatrix} f_1 \\ \vdots \\ f_m \end{bmatrix}}_{\boldsymbol{f}} + \underbrace{\begin{bmatrix} \boldsymbol{f}_{\text{P}} \\ \boldsymbol{\tau}_{\text{P}} \end{bmatrix}}_{\boldsymbol{w}} = \boldsymbol{0}, \tag{3}$$

where $\boldsymbol{f}_{\text{P}}, \boldsymbol{\tau}_{\text{P}}$ are the applied forces and torques, respectively, acting on the platform and $\boldsymbol{f}$ is the vector of the cable forces. The matrix $\boldsymbol{A}^{\mathrm{T}}$ is referred to as *structure matrix* and permits to investigate existence and quality of the workspace. A pose $(\boldsymbol{r}, \boldsymbol{R})$ is said to belong to the wrench-closure workspace if and only if there exist positive solutions for $\boldsymbol{f}$ for Eq. (3).

For the study presented here, we apply the structure matrix in a non-normalized form

$$\widehat{\boldsymbol{A}}^{\mathrm{T}} = \begin{bmatrix} \boldsymbol{l}_1 & \dots & \boldsymbol{l}_m \\ \boldsymbol{R}\boldsymbol{b}_1 \times \boldsymbol{l}_1 & \dots & \boldsymbol{R}\boldsymbol{b}_m \times \boldsymbol{l}_m \end{bmatrix} \tag{4}$$

that can be represented as

$$\widehat{\boldsymbol{A}}^{\mathrm{T}} \boldsymbol{L}^{-1} = \boldsymbol{A}^{\mathrm{T}} \quad , \tag{5}$$

where $\boldsymbol{L} = \mathrm{diag}(l_1, \dots l_m)$ is the diagonal matrix with the cable lengths l_i. Since each cable length l_i is always positive, the matrix is regular and trivial to determine. Technically speaking, the cable forces are linearly scaled by the matrix $\boldsymbol{L}^{-1}$. Therefore, one can use $\widehat{\boldsymbol{A}}^{\mathrm{T}}$ for analyzing wrench-closure workspace instead of $\boldsymbol{A}^{\mathrm{T}}$.

B. The Structure of the Workspace Boundary

Verhoeven [16], Gouttefarde [4], [6], Stump [15] and later Sheng [14] showed that the boundary of the wrench-closure workspace can be determined from algebraic expressions by evaluating the structure matrix. If the $\widehat{\boldsymbol{A}}^{\mathrm{T}}$ is used instead of $\boldsymbol{A}^{\mathrm{T}}$ in the expressions, the resulting terms are largely simplified and Sheng derived second or third order multivariate polynomials for the workspace boundary using computer algebra. Using this procedure, the analytic expressions N_i potentially bounding the translational workspace were shown [14] to be for a planar robot with $m = 4$ cables

$$N_1: \quad \det(\boldsymbol{A}_4, \boldsymbol{A}_2, \boldsymbol{A}_3) = 0 \tag{6}$$
$$N_2: \quad \det(\boldsymbol{A}_1, \boldsymbol{A}_4, \boldsymbol{A}_3) = 0 \tag{7}$$
$$N_3: \quad \det(\boldsymbol{A}_1, \boldsymbol{A}_2, \boldsymbol{A}_4) = 0 \tag{8}$$
$$N_4: \quad \det(\boldsymbol{A}_1, \boldsymbol{A}_2, \boldsymbol{A}_3) = 0 \tag{9}$$

and for a spatial robot with $m = 7$ cables

$$N_1: \quad \det(\boldsymbol{A}_7, \boldsymbol{A}_2, \boldsymbol{A}_3, \boldsymbol{A}_4, \boldsymbol{A}_5, \boldsymbol{A}_6) = 0 \tag{10}$$
$$N_2: \quad \det(\boldsymbol{A}_1, \boldsymbol{A}_7, \boldsymbol{A}_3, \boldsymbol{A}_4, \boldsymbol{A}_5, \boldsymbol{A}_6) = 0 \tag{11}$$
$$\vdots \qquad \vdots$$
$$N_6: \quad \det(\boldsymbol{A}_1, \boldsymbol{A}_2, \boldsymbol{A}_3, \boldsymbol{A}_4, \boldsymbol{A}_5, \boldsymbol{A}_7) = 0 \tag{12}$$
$$N_7: \quad \det(\boldsymbol{A}_1, \boldsymbol{A}_2, \boldsymbol{A}_3, \boldsymbol{A}_4, \boldsymbol{A}_5, \boldsymbol{A}_6) = 0 \quad , \tag{13}$$

where $\boldsymbol{A}_i$ is the i-th column of the non-normalized structure matrix $\widehat{\boldsymbol{A}}^{\mathrm{T}}$. Following the procedure presented in [14], a pose belongs to the workspace if a subset of the equations N_i have the same sign. This criterion is exploited later in Sec. IV to quickly compute the workspace. One can essentially do the same computation for the orientation workspace by substituting a constant position into the structure matrix and receive determinants that depend on the orientation parameters rather than the position. However, the analysis for the orientation workspace is different from the translation due to the different topology of $\mathbb{R}^3$ and the rotation group SO_3 and we do not tackle this problem here.

III. A SYMBOLIC-NUMERIC WORKSPACE APPROACH

A. The Concept

As shown by Verhoeven [16] and lately detailed by Sheng [14], one can describe the boundary of the constant orientation wrench-closure workspace by second or third order polynomials for the planar or spatial case, respectively. The basic approach to compute the workspace boundary is as follows. Firstly, one sets up the structure matrix of the robot. Secondly, the actual geometry parameters are substituted into the formula of the robot. Thirdly, the pose parameterization is introduced into the structure matrix. Then, one can compute symbolically the determinants. Evaluating the resulting symbolic expressions yields the desired parametric curves that are the boundary of the workspace. It is straight forward to execute the above workflow using a computer algebra system and even for the spatial case with 6×6 matrices, one can compute the determinant for a certain parameterization. However, if arbitrary geometry is assumed, the number of symbols in the CAS system becomes that large, that it cannot be handled.

To overcome this limitation, an symbolic-numeric approach is proposed in this work which is inspired by the method from Walker and Orin [17] for the equations of motion as well as by Hiller [10] for computing the Jacobian matrix of multi-body systems. In both contributions, some kind of coefficient identification scheme is employed to extract the numerical values of an equation with known structure from numerical evaluation with carefully chosen special values. Having realized that the mathematical structure of the expressions of the workspace boundary are second or third order multivariate polynomials, we can use a pose dependent formulation to compute values of N_i.

The surprising effect of this evaluation is, that one can reconstruct the full workspace boundary from only six (planar) or 20 (spatial) local evaluations of the structure matrix and its determinants to receive a closed-form parametric representation of the constant orientation wrench-closure workspace.

B. The 1R2T Case

The approach for the computation of the constant orientation representation for a robot with four cables is as follows. For the sake of simplicity, we omit in the following an additional index for the coefficients a for each equation N_i. Each boundary equation takes the form

$$N_i(x,y) = a_{xx}x^2 + a_x x + a_{yy}y^2 + a_y y + a_{xy}xy + a_0 \quad (14)$$

for a planar robot. On the other hand, one can numerically evaluate Eq. (6)–(9). The identification of the coefficients $a_{xx}, \ldots, a_0$ is done by computing the determinants for six position vectors $\boldsymbol{r} = [x,y]^{\mathrm{T}}$ following the scheme:

- compute the coefficient a_0 by evaluating the four determinants for the position vector $\boldsymbol{r} = \boldsymbol{0}$.
- compute a_{xx} and a_x from the determinants received from the position vectors $\boldsymbol{r} = [1,0]^{\mathrm{T}}$ and $\boldsymbol{r} = [-1,0]^{\mathrm{T}}$

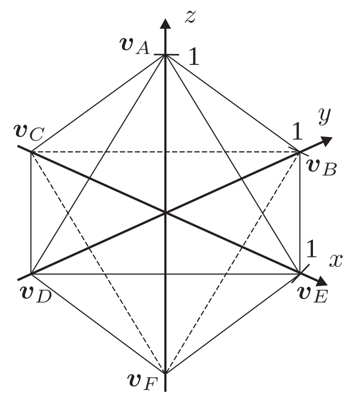

Fig. 3. Unit octahedron as initial configuration for the workspace data model

- compute a_{yy} and a_y from the determinants received from the position vectors $\boldsymbol{r} = [0,1]^{\mathrm{T}}$ and $\boldsymbol{r} = [0,-1]^{\mathrm{T}}$
- determine a_{xy} from evaluating the position $\boldsymbol{r} = [1,1]^{\mathrm{T}}$.

The numerical procedure is as follows. Compute the non-normalized structure matrix $\widehat{\boldsymbol{A}}^{\mathrm{T}}$ for the position $\boldsymbol{r} = \boldsymbol{0}$ and the desired orientation φ_0 and the respective numerical values of N_i for $i \in 1,\ldots,4$ from Eq. (6)–(9). Analyzing the polynomial expression in Eq. (14) reveals that substituting zero for both x and y cancels out all terms but the coefficient a_0 and thus $a_0 = N_i(0,0)$. Secondly, one repeats the trick to identify both a_{xx} and a_x by computing $N_i(1,0)$ and $N_i(-1,0)$. The identification of the coefficients is slightly more complicated since we have to solve a linear 2×2 equation system which coefficients are defined from our test poses $[1,0]^{\mathrm{T}}$ and $[-1,0]^{\mathrm{T}}$, thus

$$\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} a_{xx} \\ a_x \end{bmatrix} = \begin{bmatrix} N_i(1,0) - a_0 \\ N_i(-1,0) - a_0 \end{bmatrix} \quad (15)$$

has the simple solution

$$a_{xx} = \frac{1}{2}\left(N_i(1,0) + N_i(-1,0)\right) - a_0 \quad (16)$$

$$a_x = \frac{1}{2}\left(-N_i(1,0) + (N_i(-1,0)\right) \quad . \quad (17)$$

The computation of a_{yy} and a_y with the positions $[0,1]^{\mathrm{T}}$ and $[0,-1]^{\mathrm{T}}$ is done respectively. In the final step, we compute a_{xy} from the position $[1,1]^{\mathrm{T}}$ with the simple equation

$$a_{xy} = N_i(1,1) - a_{xx} - a_x - a_{yy} - a_y - a_0 \quad . \quad (18)$$

Thus, we have numerically received the exact algebraic representation of the workspace boundary by as little as computing numerically the structure matrices for six poses and determine four determinants for each structure matrix.

A prerequisite of the procedure above is that the generic test poses used in the algorithm are not singular. If the poses are singular, a rigid body transformation is applied to the parameters $\boldsymbol{a}_i, \boldsymbol{b}_i$ to move the reference points away from the singular loci.

C. The 3R3T Case

The procedure for the spatial robot is essentially the same but to avoid the tiresome computation, we turn to a linear equation formulation here. The main part in reconstructing the polynomial boundary is solving a large system. The sought polynomial boundary $N_i(x, y, z)$ takes the form

$$\begin{aligned} N_i \quad = \quad & a_{xxx}x^3 + a_{yyy}y^3 + a_{zzz}z^3 + a_{xxy}x^2y + \\ & a_{xxz}x^2z + a_{xyy}xy^2 + a_{yyz}y^2z + a_{xzz}xz^2 + \\ & a_{yzz}yz^2 + a_{xx}x^2 + a_{yy}y^2 + a_{zz}z^2 + \\ & a_{xy}xy + a_{xz}xz + a_{yz}yz + a_xx + a_yy + a_zz + \\ & a_{xyz}xyz + a_0 \end{aligned} \tag{19}$$

The system matrix for identifying the 20 coefficients of the polynomials reads as block matrix

$$\boldsymbol{S} = \begin{bmatrix} 1 & \mathbf{0}^{\mathrm{T}} & \mathbf{0}^{\mathrm{T}} & \mathbf{0}^{\mathrm{T}} & \mathbf{0}^{\mathrm{T}} & \mathbf{0}^{\mathrm{T}} & \mathbf{0}^{\mathrm{T}} & 0 \\ \mathbf{1} & \boldsymbol{A} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{1} & \mathbf{0} & \boldsymbol{A} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{1} & \mathbf{0} & \mathbf{0} & \boldsymbol{A} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{1} & \boldsymbol{C} & \boldsymbol{D} & \mathbf{0} & \boldsymbol{B} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{1} & \boldsymbol{C} & \mathbf{0} & \boldsymbol{D} & \mathbf{0} & \boldsymbol{B} & \mathbf{0} & \mathbf{0} \\ \mathbf{1} & \mathbf{0} & \boldsymbol{C} & \boldsymbol{D} & \mathbf{0} & \mathbf{0} & \boldsymbol{B} & \mathbf{0} \\ 1 & \mathbf{1}^{\mathrm{T}} & \mathbf{1}^{\mathrm{T}} & \mathbf{1}^{\mathrm{T}} & \mathbf{1}^{\mathrm{T}} & \mathbf{1}^{\mathrm{T}} & \mathbf{1}^{\mathrm{T}} & 1 \end{bmatrix}, \tag{20}$$

where the first and last column as well as the first and last row are scalars. Furthermore, $\mathbf{0}$ is a matrix with zero elements of appropriate size and $\mathbf{1}$ is a matrix having a 1 in each element. In contrast, the other columns and rows are constructed each from 3×3 matrices from the following matrices

$$\boldsymbol{A} = \begin{bmatrix} 1 & 1 & 1 \\ -1 & 1 & -1 \\ 8 & 4 & 2 \end{bmatrix} \tag{21}$$

$$\boldsymbol{B} = \begin{bmatrix} 1 & 1 & 1 \\ -1 & -1 & 1 \\ -1 & 1 & -1 \end{bmatrix} \tag{22}$$

$$\boldsymbol{C} = \begin{bmatrix} 1 & 1 & 1 \\ -1 & 1 & -1 \\ 1 & 1 & 1 \end{bmatrix} \tag{23}$$

$$\boldsymbol{D} = \begin{bmatrix} 1 & 1 & 1 \\ -1 & 1 & -1 \\ -1 & 1 & -1 \end{bmatrix} \tag{24}$$

From the structure of matrix $\boldsymbol{S}$, one can see that it is block lower triangular. Obviously, both $\boldsymbol{A}$ and $\boldsymbol{B}$ are regular, therefore also $\boldsymbol{S}$ is regular. Inverting or solving a linear system with $\boldsymbol{S}$ can be done efficiently. Basically, one can apply an algorithm similar to the procedure described in the section above by computing first the coefficient a_0, then the triple a_{xxx}, a_{xx}, a_x from a 3×3 system and so on.

The vector of the sought coefficients $\boldsymbol{k}$ of the polynomial and the right-hand side $\boldsymbol{h}$ of the equation read

$$\boldsymbol{k} = \begin{bmatrix} a_0 \\ a_{xxx} \\ a_{xx} \\ a_x \\ a_{yyy} \\ a_{yy} \\ a_y \\ a_{zzz} \\ a_{zz} \\ a_z \\ a_{xxy} \\ a_{xyy} \\ a_{xy} \\ a_{xxz} \\ a_{xzz} \\ a_{xz} \\ a_{yyz} \\ a_{yzz} \\ a_{yz} \\ a_{xyz} \end{bmatrix}, \quad \boldsymbol{h} = \begin{bmatrix} N_i(0,0,0) \\ N_i(1,0,0) \\ N_i(-1,0,0) \\ N_i(2,0,0) \\ N_i(0,1,0) \\ N_i(0,-1,0) \\ N_i(0,2,0) \\ N_i(0,0,1) \\ N_i(0,0,-1) \\ N_i(0,0,2) \\ N_i(1,1,0) \\ N_i(-1,-1,0) \\ N_i(1,-1,0) \\ N_i(1,0,1) \\ N_i(-1,0,-1) \\ N_i(1,0,-1) \\ N_i(0,1,1) \\ N_i(0,-1,-1) \\ N_i(0,1,-1) \\ N_i(1,1,1) \end{bmatrix} . \tag{25}$$

The coefficients of the polynomial can now be determined from the simple linear system

$$\boldsymbol{S}\boldsymbol{k} = \boldsymbol{h} \quad . \tag{26}$$

Computing the coefficients of the wrench-closure workspace of a spatial cable robot with seven cables thus requires the following steps:

- Numerically determine the structure matrix for the 20 positions listed in Eq. (25).
- For each of these matrices, extract the seven 6×6 determinants as described in Eq. (10) to generate the vectors $\boldsymbol{h}$.
- Solve the system Eq. (26) to compute the coefficients for each of the seven polynomials.
- The seven vectors $\boldsymbol{k}$ contain in their 140 elements the full information on the constant orientation wrench-closure workspace of the robot.

The computational costs of the main steps are: setting up 20 structure matrices, computing 140 determinants, and solving seven 20×20 linear systems, when solving the linear system can be done in linear computation time due to the almost triangular structure.

The procedure can be generalized to robots with more than seven cables, where in this case we have to compute more determinants from the structure matrices. However, the description of the exact procedure is out of the scope of this contribution.

IV. WORKSPACE COMPUTATION

Here, we compute the hull using the triangulation approach used in an earlier work [12]. One core concept is a parametric line search which can be used here in an elegant way.

Here, the translational workspace for a given orientation of the cable robot is represented by a triangulation of its hull. The idea for the determination of the workspace is to start with a unity sphere in the estimated center $\boldsymbol{m}$ of the workspace and to successively extend the sphere in radial directions. Clearly, this assumption may lead to an underestimation of the workspace and the estimation depends on the chosen value of $\boldsymbol{m}$. Contrary, for most technical applications, only robots with a compact workspace are interesting and therefore it seems reasonable to restrict a quick design procedure to such a subspace. The surface of the sphere is approximated by triangles which are created from iterative subdivision of the faces of an octahedron. Alternatively, one can also subdivide other regular polyhedrons and especially the Platonic solids with triangular facettes such as a tetrahedron or an icosahedron.

In the first step, the eight faces of an octahedron (Fig. 3) located around the point $\boldsymbol{m}$ are described as triplets of vertices, e.g. $F_1 = \{\boldsymbol{v}_\text{A}, \boldsymbol{v}_\text{B}, \boldsymbol{v}_\text{C}\}_i$. Initially, there is a set $\mathcal{L} = \{F_1, \ldots, F_8\}$ containing eight faces. These faces of the octahedron are subdivided into four congruent triangles. This is done by constructing the three vertices $\boldsymbol{v}_\text{AB}$,$\boldsymbol{v}_\text{AC}$,$\boldsymbol{v}_\text{BC}$ for each triangle F_i in $\mathcal{L}$ and projecting the generated vertices onto a unit sphere

$$\boldsymbol{v}_{ij} = \frac{\boldsymbol{v}_i + \boldsymbol{v}_j}{|\boldsymbol{v}_i + \boldsymbol{v}_j|}, \quad i, j \in \{\text{A,B,C}\}, i \neq j. \tag{27}$$

Then, the original triangle F_i is replaced by the four triangles ($\boldsymbol{v}_\text{A}$, $\boldsymbol{v}_\text{AB}$, $\boldsymbol{v}_\text{AC}$), ($\boldsymbol{v}_\text{B}$, $\boldsymbol{v}_\text{AB}$, $\boldsymbol{v}_\text{BC}$), ($\boldsymbol{v}_\text{C}$, $\boldsymbol{v}_\text{BC}$, $\boldsymbol{v}_\text{AC}$), ($\boldsymbol{v}_\text{AB}$, $\boldsymbol{v}_\text{AC}$, $\boldsymbol{v}_\text{BC}$). This process is repeated n_i times thus generating a set $\mathcal{L}$ containing $n_\text{T} = 2^{2n_i+3}$ triangles.

In the second step, the vertices of the triangles are projected onto the hull of the workspace. Starting from the estimated center $\boldsymbol{m}$ of the workspace, the line

$$L_i : \boldsymbol{r}_i = \boldsymbol{m} + \lambda_i \boldsymbol{v}_i \quad \lambda_i > 0 \tag{28}$$

is searched for its roots. Instead of the regula falsi based line search proposed in [12], one can do better with the parametric representation derived above. Since the recently used workspace criteria can only be evaluated as Boolean test of complex numerical algorithms, we used a regula falsi line search. Due to the algebraic form of the workspace boundary, we propose to substitute the line L_i in Eq. (28) into the surface constrained Eq. (14) providing the following expression which reads for the planar case

$$\begin{aligned}
&\left(a_{xx}v_{xi}^2 + a_{xy}v_{xi}v_{yi} + a_{yy}v_{yi}^2\right)\lambda_i{}^2 \\
&+ (2a_{xx}m_x v_{xi} + a_{xy}m_x v_{yi} + a_{xy}m_y v_{xi} \\
&\quad +2a_y m_y v_{yi} + a_x v_{xi} + a_y v_{yi})\,\lambda_i \\
&\quad +a_{xx}m_x^2 + a_{xy}m_x m_y + a_{yy}m_y^2 \\
&\quad +a_x m_x + a_y m_y + a_0 = 0
\end{aligned} \tag{29}$$

Analyzing this lengthy expression reveals the simple form of a quadratic equation in λ_i. Here, we earn again the benefit of the algebraic formulation since the boundary of the workspace is computed by just solving the polynomial with the well-known formula

$$\lambda_i^{1,2} = -\frac{p}{2} \pm \sqrt{\left(\frac{p}{2}\right)^2 - q} \quad . \tag{30}$$

According to the assumptions made for the hull computation, we use the smallest positive value of $\lambda_i^{1,2}$ received for any one polynomial N_i. If the roots are complex or all negative, we set $\lambda_i = 0$. In the latter case, the projection center was not part of the workspace.

For the spatial case, one can do essentially the same where the final solving for λ_i requires to compute the closed-form solution to a third order polynomial. However in both cases, we have shown that all computation steps from the geometry of the robots to the triangulation of the constant orientation workspace can be executed in closed-form with simple mathematical tools.

Even more, the triangulated hull of the workspace allows for some geometric characterizations of the workspace. It is straight forward to calculate the surface $S(\mathcal{W})$ and the volume $V(\mathcal{W})$ of the workspace as follows

$$S(\mathcal{W}) = \frac{1}{2}\sum^{\mathcal{L}} ||(\boldsymbol{r}_\text{A} - \boldsymbol{r}_\text{B}) \times (\boldsymbol{r}_\text{A} - \boldsymbol{r}_\text{C})||_2 \tag{31}$$

$$V(\mathcal{W}) = \frac{1}{6}\sum^{\mathcal{L}} ((\boldsymbol{r}_\text{A} - \boldsymbol{m}) \times (\boldsymbol{r}_\text{B} - \boldsymbol{m})) \cdot (\boldsymbol{r}_\text{C} - \boldsymbol{m}) \tag{32}$$

where $\mathcal{L}$ is the set of the triangles. For the volume, one can do better by substituting $\boldsymbol{r}_i - \boldsymbol{m} = \lambda_i \boldsymbol{v}_i$ the parametric form using the direction vector $\boldsymbol{v}_i$ and its length λ_i. Then, the equation for the volume becomes

$$V(\mathcal{W}) = \frac{1}{6}\sum^{\mathcal{L}} \lambda_\text{A}\lambda_\text{B}\lambda_\text{C}(\boldsymbol{v}_\text{A} \times \boldsymbol{v}_\text{B}) \cdot \boldsymbol{v}_\text{C}, \tag{33}$$

where the scalar value of the product $(\boldsymbol{v}_\text{A} \times \boldsymbol{v}_\text{B}) \cdot \boldsymbol{v}_\text{C}$ is equal for all triangles and depends only on the number of subdivisions n_T done. Thus, one finds the simple form

$$V(\mathcal{W}) = \frac{(\boldsymbol{n}_\text{A} \times \boldsymbol{n}_\text{B}) \cdot \boldsymbol{n}_\text{C}}{6} \sum^{\mathcal{L}} \lambda_\text{A}\lambda_\text{B}\lambda_\text{C}, \tag{34}$$

with the constant factor $V_i^{(n_\text{T})} = (\boldsymbol{n}_\text{A} \times \boldsymbol{n}_\text{B}) \cdot \boldsymbol{n}_\text{C}$. Eventually, even these expressions are received in a constant number of steps without approximation excepts for the assumption that the triangulation for the quadratic and cubic surfaces is exact. However, the polynomial form of the workspace boundary allows to compute and bound the error for the triangulation e.g. using interval analysis.

V. Computation Results

A. Quantitative Results

The workspace of the cable robot IPAnema 1 was determined for verification purpose using the algebraic expression method. The robot has seven cables and its geometrical parameters are given in Tab. I. The determined translational wrench-closure workspace is depicted in Fig. 4.

TABLE I

IPANEMA 1 GEOMETRICAL PARAMETERS FOR ROBOT WITH SEVEN CABLES: PLATFORM VECTORS $\boldsymbol{b}$ AND BASE VECTORS $\boldsymbol{a}$

cable i	platform vector $\boldsymbol{b}_i$	base vector $\boldsymbol{a}_i$
1	$[-0.125, 0.0, 0.0]^{\mathrm{T}}$	$[0.0, 0.0, 0.0]^{\mathrm{T}}$
2	$[-0.125, 0.0, 0.0]^{\mathrm{T}}$	$[4.0, 0.0, 0.0]^{\mathrm{T}}$
3	$[0.0, 0.25, 0.0]^{\mathrm{T}}$	$[4.0, 3.0, 0.0]^{\mathrm{T}}$
4	$[0.0, 0.25, 0.0]^{\mathrm{T}}$	$[0.0, 3.0, 0.0]^{\mathrm{T}}$
5	$[-0.125, 0.0, 0.0]^{\mathrm{T}}$	$[0.0, 0.0, 2.0]^{\mathrm{T}}$
6	$[-0.125, 0.0, 0.0]^{\mathrm{T}}$	$[4.0, 0.0, 2.0]^{\mathrm{T}}$
7	$[0.0, 0.25, 0.0]^{\mathrm{T}}$	$[2.0, 3.0, 2.0]^{\mathrm{T}}$

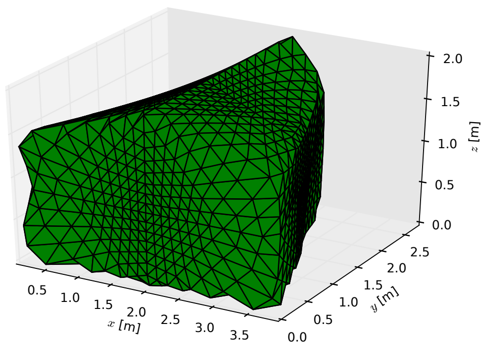

Fig. 4. Constant orientation wrench-closure workspace of the IPAnema 1 design with seven cables.

B. Computation Time

In order to determine the computational costs of the proposed method, an implementation in C++ was employed. In order to compute the matrix operations including the evaluation of the determinants, the eigen 3 library was used. The computation time was assessed on an Intel Core i5-3320M 2.6 GHz, Visual C++ 2010 using a single thread. A first test for a planar robot reveals computation times of around 0.12 ms per constant orientation evaluation with 36 points on the boundary and 0.26 ms for a resolution with 360 points. The computation time for computing the coefficients of the workspace polynomials without workspace computation are estimated to be 0.025 ms. Testing the components of the vector base of the matrix kernel to have the same sign, leads to a computation time of 10 ms for 360 poses.

The evaluation of wrench-feasibility using the fast closed-form method [13] requires for 360 points with regula falsi line search around 12.5 ms.

VI. CONCLUSIONS

In this paper, a numerical approach to compute the wrench-closure workspace of cable-driven parallel robots is proposed. The method efficiently exploits findings from theoretical considerations on the structure of the workspace where a simple but efficient numerical scheme is presented to execute the actual computation. A surprising result of the study is that for a planar cable robot with four cables the translational wrench-closure workspace can be reconstructed from the numerical evaluation of the structure matrix of only six regular poses. In the spatial case, the same procedure is possible using only 20 regular poses. The numerical algorithm is applied to some case studies showing very short computation time. Interestingly, the mathematical tools required to do the computations are limited to computing determinants and solving linear systems with standard tools. All steps are executed explicitly without numerical iteration.

Based on these results, we expect more efficient assessment of workspace computation in real-time control, e.g. to compute the translational distance between the current robot pose and the boundary of the workspace.

REFERENCES

[1] James S. Albus, Roger V. Bostelman, and Nicholas G. Dagalakis. The NIST robocrane. *Journal of Research at the National Institute of Standards and Technology*, 97(3):373–385, 1992.

[2] Kaveh Azizian, Philippe Cardou, and Brian Moore. On the boundaries of the wrench-closure workspace of planar parallel cable-driven mechanisms. In *Proceedings of the ASME Design Engineering Technical Conference*, 2(PARTS A AND B):203–212, 2010.

[3] Kaveh Azizian, Philippe Cardou, and Brian Moore. Classifying the boundaries of the wrench-closure workspace of planar parallel cable-driven mechanisms by visual inspection. *Journal of Mechanisms and Robotics*, 4(2), 2012.

[4] Marc Gouttefarde and Clément Gosselin. On the properties and the determination of the wrench-closure workspace of planar parallel cable-driven mechanisms. *Proceedings of the ASME Design Engineering Technical Conference*, 2:337–346, 2004.

[5] Marc Gouttefarde and Clément Gosselin. Wrench-closure workspace of six-dof parallel mechanisms driven by 7 cables. *Transactions of the Canadian Society for Mechanical Engineering*, 29(4):541–552, 2005.

[6] Marc Gouttefarde and Clément Gosselin. Analysis of the wrench-closure workspace of planar parallel cable-driven mechanisms. *IEEE Transactions on Robotics*, 22(3):434–445, 2006.

[7] Marc Gouttefarde, Jean-Pierre Merlet, and David Daney. Determination of the wrench-closure workspace of 6-dof parallel cable-driven mechanisms. In *Advances in Robot Kinematics (ARK)*, pages 315–322, Ljubljana, Slovenia, 2006. Springer-Verlag.

[8] Hamoon Hadian and Abbas Fattah. Best kinematic performance analysis of a 6-6 cable-suspended parallel robot. In *IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications, MESA 2008*, pages 510–515, 2008.

[9] Mahir Hassan and Amir Khajepour. Analysis of a large-workspace cable-actuated manipulator for warehousing applications. In *Proceedings of the ASME International Design Engineering Technical Conferences & Computers and Information in Engineering Conference, DETC2009*, 7(PART A):45–53, 2010.

[10] Manfred Hiller and Andrés Kecskeméthy. Equations of motion of complex multibody systems using kinematical differentials. In *Proceedings 9th Symposium on Engineering Applications of Mechanics*, pages 425–430, Ontario, Canada, 1988.

[11] Aiguo Ming and Toshiro Higuchi. Study on multiple degree-of-freedom positioning mechanism using wires (part 1). *International Journal of the Japan Society for Precision Engineering*, 28(2):131–138, 1994.

[12] Andreas Pott. Forward kinematics and workspace determination of a wire robot for industrial applications. In *Advances in Robot Kinematics (ARK)*, pages 451–458, Baz-sur-Mer, France, 2008. Springer-Verlag.

[13] Andreas Pott, Tobias Bruckmann, and Lars Mikelsons. Closed-form force distribution for parallel wire robots. In *Computational Kinematics*, pages 25–34. Springer, 2009.

[14] Zhiyu Sheng, Joon-Hyuk Park, Paul Stegall, and Sunil Kumar Agrawal. Analytic determination of wrench closure workspace of spatial cable driven parallel mechanisms. In *ASME 2015 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference*. 2015.

[15] Ethan Stump and Vijay Kumar. Workspaces of cable-actuated parallel manipulators. *Journal of Mechanical Design*, 128(1):159–167, 2006.

[16] Richard Verhoeven. *Analysis of the Workspace of Tendon-based Stewart Platforms*. PhD thesis, University of Duisburg-Essen, 2004.

[17] M. Walker and D. Orin. Efficient dynamic computer simulation of robotic mechanisms. *Journal of Dynamical Systems, Measurement and Control*, 104, 205–211, 1982.