T. Bruckmann/L. Mikelsons/M. Hiller/D. Schramm

Analysis and Synthesis of Parallel Tendon-based Manipulators

1 Introduction

Parallel cable-driven Stewart-Gough platforms consist of a movable end-effector which, is connected to a machine frame by motor driven cables. At the Chair of Mechatronics, a testbed for tendon-based Steward-platforms (SEGESTA, Seilgetriebene Stewart-Plattformen in Theorie und Anwendung) has been developed during the past few years (see Fig. 1a). Presently, the SEGESTA teststand has $n = 6$ d.o.f. and uses $m = 7$ tendons to move the platform along desired trajectories [3]. In a future modified version of SEGESTA it is planned to add an eighth tendon. Since cables can transmit only tension forces, at least $m = n + 1$ cables are needed to tense a system having n degrees-of-freedom. This results in a kinematical redundancy and gives $m - n$ degrees-of-freedom in the cable force distribution. For this reason, the workspace analysis is complex and very time-consuming. Therefore, reliable and robust algorithms are demanded to calculate the resulting workspace for a given parameter set (cable winch positions and platform connection points). To analyze the workspace, discrete methods are widely used. However, the drawback of these methods is that intermediate points on the discrete calculation grids are neglected. Especially for parallel mechanisms, this may lead to false results and thus be dangerous. A promising way to avoid this kind of discretization problems may be analyzing the workspace by means of so-called interval analysis. Within this paper, the workspace is defined by a force equilibrium and the workspace analysis is demonstrated for this criterion. The force equilibrium can be described as (see see Fig. 1b):

$$\begin{bmatrix} \boldsymbol{v}_1 & \cdots & \boldsymbol{v}_m \\ \boldsymbol{p}_1 \times \boldsymbol{v}_1 & \cdots & \boldsymbol{p}_m \times \boldsymbol{v}_m \end{bmatrix} \begin{bmatrix} f_1 \\ \vdots \\ f_m \end{bmatrix} + \begin{bmatrix} \boldsymbol{f}_p \\ \boldsymbol{\tau}_p \end{bmatrix} = 0 \tag{1}$$

with $\boldsymbol{v} = \frac{\boldsymbol{l}_i}{|\boldsymbol{l}_i|}$ and $\boldsymbol{f} > 0$ or in a more compact form as

$$\boldsymbol{A}^{\mathrm{T}}\boldsymbol{f} + \boldsymbol{w} = 0, \quad \boldsymbol{f} > 0. \tag{2}$$

Using this formula as a criterion for workspace calculation, the problem can be described as a *Constraint Satisfaction Problem (CSP)*. A possible algorithm to solve CSPs bases on interval analysis.

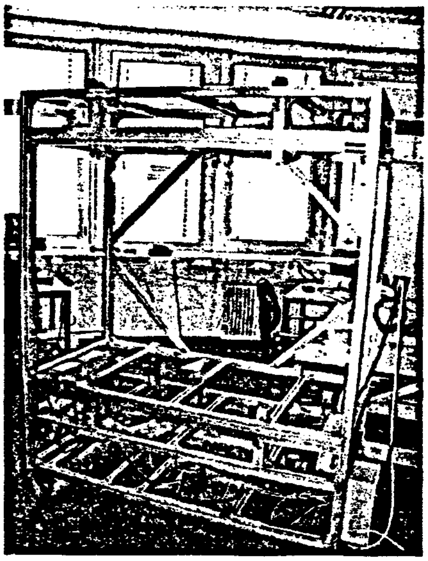

Fig. 1a SEGESTA Testbed

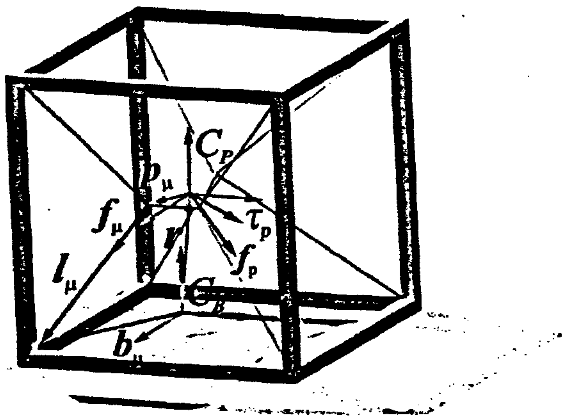

Fig. 1b Symbol Definitions for a Tendon-Based Stewart-Platform

2 Interval Analysis

For two real numbers a and b an interval $I = [a, b]$ is defined as follows

$$[a,b] := \{r \mid a \leq r \leq b\} \text{ where } a < b. \tag{3}$$

Then b is called the supremum and a the infimum of I. A n-tupel of intervals is called box or interval vector. It is possible to define every operation $\circ$ on $\mathbf{R}$ on the set of intervals $\mathbf{I} = \{[a,b] \mid a, b \in \mathbf{R}\}$, such that the following holds:
Let $I_0, I_1 \in \mathbf{I}$ be two intervals. Then

$$\forall u \in I_0,\ \forall v \in I_1\ \exists z \in I_0 \circ I_1 \tag{4}$$

where

$$z = u \circ v. \tag{5}$$

Hence

$$\max_{u \in I_0, v \in I_1} u \circ v \leq \mathrm{Sup}(I_0 \circ I_1), \tag{6}$$

where $<$ is not unusual. This phenomenon is called overestimation and causes additional numerical effort to get sharp boundaries. For sure the same holds for min and Inf. Thus for input intervals $I_0, \ldots, I_n$ interval analysis delivers evaluations for the domain $I_0 \times I_1 \times \ldots \times I_n$. This evaluation is guaranteed to include all possible solutions, e.g.

$$[1,3] + [2,4] = [3,7]. \tag{7}$$

In some cases overestimation can be eliminated, e.g.

$$[1,3]+[1,3]\cdot[-2,1]=[-5,6] \tag{8}$$

while

$$[1,3]\cdot(1+[-2,1])=[-3,6]. \tag{9}$$

3 Solving CSPs using Interval Analysis

A CSP is the problem of determining all $c \in X_c$ such that

$$\begin{array}{r} \Phi(c,v)>0 \\ \forall\, v \in X_v \end{array} \tag{10}$$

where Φ is a system of real functions defined on a real domain representing the constraints. It will be shown later that for a description of the workspace, this problem has to be extended to

$$\begin{array}{r} \Phi(c,v,e)>0 \\ \forall\, v \in X_v \\ \exists\, e \in X_e. \end{array} \tag{11}$$

Within this definition

- c is the vector of the calculation variables,
- v is the vector of the verification and,
- e is the vector of the existance variables.

Additionally, there may exist a vector of (e.g. geometrical) parameters g. The solution set for calculaton variables of a CSP is called X_S i.e.

$$\begin{array}{r} \Phi(c,v,e)>0 \\ \forall\, c \in X_S \subset X_c \\ \forall\, v \in X_v \\ \exists\, e \in X_e. \end{array} \tag{12}$$

As shown in detail in [6], the CSP can be solved using interval analysis, which guarantees reliable solutions [2],[5],[4]. Solving the CSP with interval analysis delivers a list of boxes $\mathcal{L}_S$ representing an inner approximation of X_S. According to eqn. 12, the solutions in $\mathcal{L}_S$ hold for total X_v and a subset of X_e. Additionally, available implementations for interval analysis computations are robust against rounding effects. General CSP solving algorithms have been proposed in [6], which have to be changed slightly:

Algorithm *CSP Solver*
Given a system of inequations $\Phi(b(c,v,e))$ as a CSP. Set a desired precision ϵ for the calculation.

1. Define a start box b_0.
2. Evaluate $\Phi(b_i)$ and analyze the result:
3. If $\inf(\Phi(b_i)) > 0$, entire box belongs to the solution, mark as valid.
4. If $\sup(\Phi(b_i)) < 0$, entire box is not part of the solution, mark as invalid.
5. Otherwise, result is undefined. Bisect box, if the diameter is bigger than ϵ and store the new parts.
6. For Calculation, store all valid boxes. For Verification, exit with invalid when any box is invalid. For Existence check, exit with valid when any box is valid.
7. If unprocessed boxes remain, goto step 2 and investigate next unkown box.

4 Continuous Workspace Analysis

Examining eqn. 2, the structure matrix A^T needs to be inverted to calculate the tendon forces f from a given platform pose and given external forces w. Since A^T has a non-squared shape, this is usually done using the Moore-Penrose pseudo inverse. Thus, the calculated forces will be a least squares solution. In fact, not a least squares result but a force distribution within predefined tensions is demanded. To overcome this problem, the structure matrix is divided into a squared matrix A_{pri}^T and a second matrix A_{sec}^T with $m - n$ columns. Now, the resulting force distribution can be calculated as

$$f_{pri} = -A_{pri}^{T\ -1}(w + A_{sec}^T f_{sec}). \tag{13}$$

In this equation, f_{sec} is unknown. Every point and wrench satisfying

$$f_{min} \leq f_{sec} \leq f_{max} \tag{14}$$

and leading to primary tendon forces

$$f_{min} \leq f_{pri} \leq f_{max} \tag{15}$$

belongs to the workspace. Hence eqns. 14 and 15 represent a CSP of the form of eqn. 11 with f_{sec} as existence variable. In case of a single redundancy ($r = 1$), an arbitrary column of A^T can be chosen to be A_{sec}^T [7], but by experiment it was found out that it is more efficient to loop through the columns than to fix one column to get the same precision. Fixing one column forces the algorithm to divide the boxes down to a very small size until the final result is at hand. This may lead to a higher computational effort than to loop through the columns. On the other hand, for $r \geq 2$, looping is mandatory, i.e. all permutations have to be tested until a final result can be set. To calculate a workspace for a specific robot, the following variable set for the CSP is used:

- The platform coordinates are the *calculation variables*.

- The tendon forces f_{sec} are the *existence variables*. Using intervals, this can be done in a "natural" way by using the tendon force limits as the interval boundaries.

- Optionally, the exerted external wrench w and desired platform orientations can be set as *verification variables*. Workspace for a fix orientation of the platform is called *constant orientation workspace*. On the other hand, sometimes free orientation of the platform within given ranges must be possible within the whole workspace. The resulting workspace is called the *total orientation workspace.*

The algorithm determines all boxes, which are guaranteed to be reachable with at least one valid force distribution, for all desired orientations and external wrenches.

5 Continuous Workspace Synthesis

Workspace synthesis describes the process to obtain the geometrical parameters for a set $\mathcal{M} := \{m_\alpha\}, \alpha \in \mathcal{I}$ of manipulators providing a predefined workspace. This predefined workspace $\mathcal{W}_p$ is guaranteed to be a subset of each obtained manipulator's workspace $\mathcal{W}_\alpha$, i.e.

$$\mathcal{W}_p \subset \mathcal{W}_\alpha. \tag{16}$$

In order to formulate the synthesis problem as a CSP of the form 11, eqns. 14 and 15 are considered again. This time the roles of the variables have to be interchanged:

- The workspace coordinates are the *verification variables*, i.e. all found parameter sets m_α describe manipulator configurations having at least the demanded workspace $\mathcal{W}_p$. Again, optionally the exerted external wrench w and a platform orientation can be given as *verification variables*.

- The tendon forces f_{sec} are the *existence variables*

- The geometrical parameters are the calculation variables, i.e. $\mathcal{M} = \mathcal{X}_S$.

Solving this CSP with the described algorithm leads to a solution set represented as a list of boxes $\mathcal{L}_s \subset \mathcal{X}_S$ containing the geometrical parameters of the manipulators. Note that the following holds

$$\lim_{\epsilon \to 0} \mathcal{L}_s \backslash \mathcal{X}_S = \emptyset, \tag{17}$$

where

$$\epsilon = \begin{bmatrix} \epsilon_c \\ \epsilon_v \\ \epsilon_e \end{bmatrix}. \tag{18}$$

In general $\mathcal{L}_s$ will be a real subset of $\mathcal{X}_S$, but due to eqn. 17 $\mathcal{X}_S$ can be approximated by $\mathcal{L}_s$ up to any precision by reducing ϵ. Usually a synthesis requires the analysis of many possible manipulator configurations, thus it requires a multiple of the computation time used in the analysis process.

6 Manipulator Design Optimization

Often, in industrial applications the term "optimal" is used with respect to economic aspects, i.e. costs. In the case of tendon-based parallel robots, the most cost-driving factor are the tendon winch units. Thus, every optimization method should reduce the number of winches as far as possible. Subsequently, within the set of all suitable robots with a minimum number of winches, a further optimization criterion can be employed. A reasonable choice is the volume expansion. On one hand, reducing the expansion of the robot saves space within a production facility, which reduces costs, on the other hand, the required tendon lengths are minimized. Since the usage of modern high-tech tendons is convenient in terms of safety, reliability and performance, the costs for tendons can be remarkable. In literature, usually the optimization is performed with respect to the size (or volume) of the workspace, e.g. [1]. Here, another approach is used [6]: Not a maximum size of the workspace is demanded, but the guaranteed covering of a predefined domain taking the above mentioned economic aspects into account at the same time. The following algorithm performs the required steps:

Algorithm *Design Optimization*

1. Set the number of tendons $m = 3$
2. Perform a Workspace Synthesis
3. If the Synthesis delivers no solution, $m = m + 1$ and go to 2
4. Perform a Global Optimization on the solution set regarding a cost function $C(c)$

Note that the initial number of tendons was chosen with respect to practical aspects, i.e. d.o.f. = 2. The Global Optimization can be performed using Interval Analysis again and is described in detail in [2].

7 Conclusion

The workspace analysis for parallel cable-driven Stewart-Gough platforms is complex and very time-consuming. Reliable and robust algorithms are demanded to calculate the resulting workspace for a given parameter set (cable winch positions and platform connection points). However, the drawback of discrete methods is that intermediate points on the discrete calculation grids are neglected. Especially for parallel mechanisms, this may lead to false results and thus be dangerous. In this paper, discretization problems are avoided by analyzing the workspace by so-called interval arithmetics. The presented algorithms yield guaranteed workspaces and are usable for both the completely restrained systems ($m = n + 1$) as well as for the more difficult redundantly restrained systems ($m > n + 1$). For practical application, e.g. in robotics, parameter synthesis to generate desired workspace geometries is mandatory. However, in general the resulting workspace geometries are very complex and not intuitive to the design engineer. Here, the extension of the analysis methods to methods usable for the synthesis is shown.

Acknowledgements

This work is supported by the German Research Council (Deutsche Forschungsgemeinschaft) under HI370/24-1.

References

[1] Hay A.M. and Snyman J.A. The optimal synthesis of parallel manipulators for desired workspace. In *Advances in Robot Kinematics*, pages 337–346, Caldes de Malavalla, 2002.

[2] Eldon Hansen. *Global Optimization using Interval Analysis*. Marcal Dekker, Inc., 1992.

[3] M. Hiller, S. Fang, S. Mielczarek, R. Verhoeven, and D. Franitza. Design, analysis and realization of tendon-based parallel manipulators. *Mechanism and Machine Theory*, 40, 2005.

[4] J.-P. Merlet. A generic trajectory verifier for the motion planning of parallel robots. *Journal of Mechanical Design*, 123:510–515, 2001.

[5] J.-P. Merlet. Solving the forward kinematics of a gough-type parallel manipulator with interval analysis. *Int. J. of Robotics Research*, 23(3):221–236, 2004.

[6] A. Pott. *Analyse und Synthese von Parallelkinematik-Werkzeugmaschinen*. Ph. D. dissertation, Gerhard-Mercator-University, Duisburg, Germany, 2007. To be published.

[7] R. Verhoeven. *Analysis of the Workspace of Tendon-Based Stewart-Platforms*. Ph. D. dissertation, Gerhard-Mercator-University, Duisburg, Germany, 2004. Available under http://deposit.ddb.de/.

Authors:

Tobias Bruckmann
University of Duisburg-Essen, Lehrstuhl für Mechatronik
Lotharstraße 1, 47057 Duisburg
Tel.: +49 (0) 203 / 379-1908, Fax: +49 (0) 203 / 379-4494, eMail: bruckmann@imech.de

Lars Mikelsons
University of Duisburg-Essen, Lehrstuhl für Mechatronik
Lotharstraße 1, 47057 Duisburg
Tel.: +49 (0) 203 / 379-1908, Fax: +49 (0) 203 / 379-4494, eMail: mikelsons@imech.de

Dieter Schramm
University of Duisburg-Essen, Lehrstuhl für Mechatronik
Lotharstraße 1, 47057 Duisburg
Tel.: +49 (0) 203 / 379-3275, Fax: +49 (0) 203 / 379-4494, eMail: schramm@imech.de

Manfred Hiller
University of Duisburg-Essen, Lehrstuhl für Mechatronik
Lotharstraße 1, 47057 Duisburg
Tel.: +49 (0) 203 / 379-3337, Fax: +49 (0) 203 / 379-4494, eMail: hilller@imech.de