Interval Methods for Model Qualification: Methodology and Advanced Application

Julien Alexandre Dit Sandretto, Gilles Trombettoni, David Daney

Interval Methods for Model Qualification: Methodology and Advanced Application

Julien Alexandre dit Sandretto
Gilles Trombettoni
David Daney

RESEARCH REPORT
N° 7880
Janvier 2012
Project-Teams COPRIN

ISSN 0249-6399 ISRN INRIA/RR--7880--FR+ENG

Interval Methods for Model Qualification: Methodology and Advanced Application

Julien Alexandre dit Sandretto*
Gilles Trombettoni†
David Daney‡

Project-Teams COPRIN

Research Report n° 7880 — Janvier 2012 — 20 pages

Abstract: It is often too complex to use, and sometimes impossible to obtain, an actual model in simulation or command field . To handle a system in practice, a simplification of the real model is then necessary. This simplification goes through some hypotheses made on the system or the modeling approach. In this paper, we deal with all models that can be expressed by real-valued variables involved in analytical relations and depending on parameters. We propose a method that qualifies the simplification validity by verifying a quality threshold on the hypothesis relevance. This method, based on interval analysis, can check the acceptance of the hypothesis in a full range of the whole model space, and can give bounds to the quality threshold and to the model parameters.Our approach is experimentally validated on a robotic application.

Key-words: Modelling, Simplification, Interval analysis, Cable-driven Robot

* Inria
† Inria,I3S
‡ Inria

RESEARCH CENTRE
SOPHIA ANTIPOLIS – MÉDITERRANÉE

2004 route des Lucioles - BP 93
06902 Sophia Antipolis Cedex

Méthodes intervalles pour la qualification de modèles simplifiés rapport de recherche Inria

Résumé : Ce document presente une methode base sur l'analyse par intervalle servant a la verification d'une hypothese utilisee dans le cadre d'une simplification de modele. Une application en robotique est presentee.

Mots-clés : Modèlisation, Simplification, Analyse par intervalles, Robot à câbles

Contents

1 Introduction

The level of realism of a model representing a (mechanical, financial, chemical or whatever) system must be generally adjusted to be enough accurate and realistic but also sufficiently simplified to be evaluated, manipulated, or used in a simulation/control application. Some methods can be exploited to reduce the modeling complexity as statistical procedures, mathematical analysis, systemic analysis, etc. These simplifications are always based on assumptions, e.g., setting some parameters to zero. The necessity of model simplification is well-known but the validation of the simplified model is rarely achieved. Few researches have been achieved on checking the validity of model simplification. Let us mention Pacut and Kolodziej who consider a simplified model acceptable if the discrepancy between this model and the reference model could be identified as a random error. They check it with a statistical test in [12]. Moreover, an hypothesis used to simplify a model should be acceptable for all the possible model cases of use. In other words, the simplified model needs to be "close" to reality for all feasible values of model entries.

Interval analysis, because of its set-oriented approach, allows us to evaluate a function for all feasible values (generally bounded) of a variable [10, 11]. Indeed, interval analysis (IA) [5, 10, 11, 2] can handle the whole continuous space, contrarily to approaches based on discretization that ignore some values. In robotics, interval analysis is used to manipulate bounded uncertainties, or to consider the whole workspace of a robot.

For this capability, we first propose in this paper a method using IA to verify the validity of an hypothesis for simplifying a model. A second IA-based method can improve the knowledge, and thus the limits, of the model through a global optimization process. We finally demonstrate the feasibility of our method on a real complex robotic problem. This experimentation concerns the simplification of the cable model in the kinematics of parallel cable driven robots through the mass-less and non-elasticity hypothesis.

Overall, we present in this paper a quasi-generic approach to qualify a relevance hypothesis in a model expressed by constraints.

2 Quality of model simplification

There exist several types of models requiring a simplification, including:

- model handling that is too time-consuming, e.g. in a real-time command;
- models with a critical sub-part;
- non-linear models which need to be linearized...

In the paper, the model, or a sub-part of the model, is defined as follows: A model M gives n values ν, function of m entries e and p parameters ρ:

$$M(e, \rho) = \nu \quad (1)$$

The entries e can take every values in the set Σ^m, denoted by Σ below, which defines the possible area of model use. Eq. (1) must have an analytical form, such as a polynomial system, a trigonometric equation, or a constraint system.

2.1 A model under hypothesis

If handling a model M is too time-consuming, it should be useful to simplify it. As said before, some methods exist in this aim. However, the most used and easy to obtain simplification method is the system analysis. For example, if the model contains a hard to compute part which helps to express a negligible phenomenon in comparison to the main system, this part can be reduced to a small constant or zero.

The reduced model obtained by simplification of M could be written M_h and we have: $M(e, \rho) \neq M_h(e_h, \rho_h)$. (In the simplified model, the entries e_h and the parameters ρ_h are sub-sets of e and ρ, so that the sets of entries $e - e_h$ and of parameters $\rho - \rho_h$ are not used anymore in the simplified model.)

2.2 Validity of hypothesis

The hypothesis is often considered "acceptable" if M is close to M_h in the sense of a distance derived from ν for all possible entries.

$$\forall e \in \Sigma : M(e, \rho) \simeq M_{hypothesis}(e_h, \rho_h) \tag{2}$$

The notion of acceptance or validity is intrinsically expressed by a threshold. In fact, the last equation could be written more formally as:

$$\forall e \in \Sigma : Dist(M(e, \rho) - M_{hypothesis}(e_h, \rho_h)) \leq \epsilon \tag{3}$$

Where $Dist()$ represents a distance form such as, for example, an Euclidean norm, an infinite norm or an absolute value.

Rewritten as a constraint system, the equations (1) and (3) become: $\forall e \in \Sigma$:

$$\begin{cases} M(e, \rho) = \nu \\ Dist(\nu - M_h(e_h, \rho_h)) \leq \epsilon \end{cases} \tag{4}$$

Quality threshold

A difficulty in this representation is the value of the threshold. It could be linked to the model use or domain of application. For instance, the accuracy expected for the model could constitute a good threshold because it is useless to provide information under this accuracy. The computation precision could also be chosen to obtain a threshold. The accuracy of instruments used in a measurement process provides a threshold as well.

2.3 Model analysis

In the previous section, a constraint system has been defined to express a hypothesis validity. In addition to this yes-or-no approach, we can compute bounds of parameters for better evaluating the simplified model. One way is to find the bounds of some parameters which guarantee the validity of the model simplification for all considered entries. It is mathematically equivalent to find the domain of parameters Φ such that

$$\forall \rho \in \Phi, \forall e \in \Sigma : Dist(M(e, \rho) - M_h(e_h, \rho_h)) \leq \epsilon \tag{5}$$

Also, during an analysis of the model, finding a subset $\hat{\Sigma} \subset \Sigma$ may add a powerful information for reducing the space of model use. This information is described by: Find $\hat{\Sigma}$ such that

$$\forall e \in \hat{\Sigma} : Dist(M(e,\rho) - M_h(e_h,\rho_h)) \leq \epsilon \tag{6}$$

Finally, a dual analysis can be to compute, and not check, the guaranteed bounds ϵ of $Dist(M(e,\rho) - M_h(e_h,\rho_h)), \forall e \in \Sigma$. Given a value set Σ, this analysis computes a guaranteed bound ϵ_{max} such that:

$$Dist(M(e,\rho) - M_h(e_h,\rho_h)) \leq \epsilon_{max} \tag{7}$$

Two bounds could also be computed:

$$\epsilon_{min} \leq M(e,\rho) - M_h(e_h,\rho_h) \leq \epsilon_{max} \tag{8}$$

The equation (8) provides an additional information. If $\epsilon_{min} > 0$, then the simplified model is never equal to the initial model (same if $\epsilon_{max} < 0$).

3 Development in terms of sets

Our problematics is firstly to verify this hypothesis for all entries in Σ to bring the guarantee that the simplification is valid.

For this purpose, we express the errors:

$$\begin{aligned} \sigma &= Dist(M(e,\rho) - M_h(e_h,\rho_h)) \\ &= Dist(\nu - \nu_h) \end{aligned}$$

made between the most realistic model and the simplified model.

We will have to verify that, for all $e \in \Sigma$, this error lies under an acceptable threshold ϵ: $\sigma \leq \epsilon$. Therefore, we can first define the subset S_e of acceptable entries:
$S_e = \{e \in \Sigma : \sigma \leq \epsilon\}$.

3.1 Checking the hypothesis validity

The proposed verification consists in checking the hypothesis for all the entries Σ.We want thus to prove that $S_e \equiv \Sigma$. Since it is difficult to characterize this infinite set, we resort to the dual set $S_{!e} = \{e \in \Sigma : \sigma > \epsilon\}$. We remark that $S_e \cup S_{!e} \equiv \Sigma$.

Proving that $S_{!e}$ is empty (i.e the dual system has no solution) implies indeed that the hypothesis $\sigma \leq \epsilon$ is verified for every model entries.

3.2 Qualifying the simplified model

The sets S_e and $S_{!e}$ give an answer about the validity of the hypothesis. However, we could expect additional quantified information, defined through the computation of bounds, such as:

- The minimal (and/or maximal) parameters ρ_{min} (and/or ρ_{max}) value satisfying the hypothesis, expressed by $if \rho < \rho_{min}$: $S_{!e} \neq \emptyset$
- The maximal error committed in Σ, defined by $S_\sigma = Max(Dist(M(e,\rho) - M_{hypothesis}(e_h,\rho_h)))$, $\forall e \in \Sigma$

4 Background about intervals

The problematics demands a robust solver which could consider a whole Σ made of an infinity of points and give a reliable result. In addition, the system to be solved may be non-linear and difficult.

Interval analysis meets these requirements by using algorithmic principles exploiting constraints and sub-spaces containing an infinity of points, without risk of solution loss.

4.1 Basics of interval arithmetic

An interval $[x_i] = [\underline{x_i}, \overline{x_i}]$ defines the set of reals x_i s.t. $\underline{x_i} \leq x_i \leq \overline{x_i}$. $\mathbb{IR}$ denotes the set of all intervals. The size or width of $[x_i]$ is $w([x_i]) = \overline{x_i} - \underline{x_i}$. A **box** $[x]$ is the Cartesian product of intervals $[x_1] \times ... \times [x_i] \times ... \times [x_n]$. Its width is defined by $\max_i w([x_i])$.

Interval arithmetic [10] extends to $\mathbb{IR}$ elementary functions over $\mathbb{R}$. For instance, the interval sum (i.e., $[x_1]+[x_2] = [\underline{x_1}+\underline{x_2}, \overline{x_1}+\overline{x_2}]$) encloses the image of the sum function over its arguments, and this enclosing property basically defines what we call an *interval extension*.

Definition 1 (Extension of a function to $\mathbb{IR}$)
Consider a function $f : \mathbb{R}^n \to \mathbb{R}$.
$[f] : \mathbb{IR}^n \to \mathbb{IR}$ is said to be an **extension** *of f to intervals if:*

$$\begin{array}{lll} \forall [x] \in \mathbb{IR}^n & [f]([x]) \supseteq \{f(x),\ x \in [x]\} \\ \forall x \ \in \mathbb{R}^n & f(x) = [f](x) \end{array}$$

In our context, the expression of a function f is always a composition of elementary functions. The **natural extension** $[f]_N$ is then simply a composition of the corresponding interval operators. The **Taylor extension** uses the first or second order Taylor development of the function and computes its natural extension.

Example Consider $f(x_1, x_2) = 3x_1^2 + x_2^2 + x_1 * x_2$ in the box $[x] = [-1, 3] \times [-1, 5]$. The natural evaluation provides: $[f]_N([x_1], [x_2]) = 3 * [-1, 3]^2 + [-1, 5]^2 + [-1, 3] * [-1, 5] = [0, 27] + [0, 25] + [-5, 15] = [-5, 67]$. The partial derivatives are: $\frac{\partial f}{\partial x_1}(x_1, x_2) = 6x_1 + x_2$, $[\frac{\partial f}{\partial x_1}]_N([-1, 3], [-1, 5]) = [-7, 23]$, $\frac{\partial f}{\partial x_2}(x_1, x_2) = x_1 + 2x_2$, $[\frac{\partial f}{\partial x_2}]_N([x_1], [x_2]) = [-3, 13]$. The interval first-order Taylor evaluation with $\dot{x} = (1, 2)$ yields: $[f]_T([x_1], [x_2]) = 9 + [-7, 23] * [-2, 2] + [-3, 13] * [-3, 3] = [-76, 94]$.

Definition 2 (Overestimation of a set)
Consider the set $F = \{f(x),\ x \in [x]\}$, the interval extension $[f]([x])$ is an overestimation of F denoted by:
$\Box F = [f]([x])$.

4.2 Interval methods for constraint satisfaction and optimization

4.2.1 Interval methods for solving a constraint system

Interval methods can accurately approximate by boxes the set of solutions of a constraint system. The solving process starts from an initial box representing the search space and builds a search tree, following a *Branch & Contract* scheme:

- *Branch:* the current box is **bisected** on one dimension (variable), generating two sub-boxes.
- *Contract:* filtering (also called *contraction*) algorithms reduce the bounds of the box with no loss of solution.

The process terminates with *atomic* boxes of size at most ϵ on every dimension. Contraction algorithms comprise interval Newton-like algorithms issued from the numerical interval analysis community [10] along with algorithms from constraint programming.

4.2.2 Constrained optimization

Interval methods can also deal with a more difficult problem, constrained optimization, in which a solution must be found that minimizes an *objective* function while satisfying the set of constraints. To do so, the strategy follows a branch and bound schema [3]. At each iteration, the algorithm selects in the list a box $[x]$. It chooses a branching variable $x_i \in x$ heuristically, bisects $[x_i]$ and applies the main Contract&Bound procedure on the two sub-boxes. In addition to the contraction phase mentioned above, the procedure *Contract & Bound* resorts to a *lower bounding* phase and an *upper bounding* phase.

The lower bounding consists in finding a point whose cost is worse (although generally non feasible, leaving some constraints unsatisfied) than that of all the points in the studied box. To do so, linearization techniques approximates the solution set and a Simplex algorithm finds the best point in the over-estimated polytope. We call *lb* the minimum value of the lower bounds of the different boxes managed by the optimization strategy.

Also, *ub* (for upper bound) is the cost of the current best feasible point (i.e., a point satisfying the constraints) ever found during the search. This upper bounding phase is achieved by local search techniques or more sophisticated methods [13].

The search terminates when $ub - lb$ reaches a precision ϵ_{obj}.

4.3 Add-ons

For improving the contraction, our tool uses two recent algorithms. The first one is a sophisticated *constraint propagation* algorithm called `Mohc` [1]. The core of constraint propagation is to contract a box by considering a single constraint at a time, then propagating the reduction to the others. The main procedure of `Mohc` improves the state-of-the-art by better contracting the box when the handled function is monotonic w.r.t. some variables in the box. The contraction is even optimal (modulo the floating-point round-offs) when the function is monotonic w.r.t. every variable (occurring several times in the function).

`Mohc` is used in our tool as a sub-contractor of the `3BCID` algorithm [14], a variant of `3B` [7]. `3B` uses a refutation principle that splits an interval into slices. A slice at the bounds of an interval is discarded, thus contracting the box, if calling the sub-contractor (here, `Mohc`) on the resulting sub-problem leads to no solution. This process leaves generally left-side and right-side boxes that are not eliminated by the sub-contractor, and thus a "central" remaining interval. An additional role of `3BCID` is to achieve a final call to the sub-contractor on this central interval, and the (hulled) union of the three boxes is returned. Therefore a contraction may be achieved in several dimensions.

The order in which the variables are selected for the branching is also crucial. We have used with success a variant of the `smear`-based heuristic [6] described in [13]. Without detailing, a variable x will be more likely/often selected and split if its current interval $[x]$ is large and if the functions involving x in the constraints have significant partial derivatives w.r.t. x in the current box.

5 Hypothesis verification using interval approach

The set-based development described in Section 3 is directly translatable in interval analysis approach. The IA methods presented in the interval background permit to handle the constraint system (4), to verify the model simplification and to analyze the quality of the hypothesis.

5.1 Checking the hypothesis with IA

The interval methods introduced in Section 4.2.1 allow us to compute a superset $\Box S_{!e}$ of $S_{!e}$ (due to interval overestimation) by using the dual of the constraint system (4) yielded by:

$$\begin{cases} M(e,\rho) = \nu \\ Dist(\nu - M_h(e_h,\rho_h)) > \epsilon \end{cases} \tag{9}$$

Interval analysis provides the yes-or-no answer useful to validate the hypothesis made for simplifying the model M. Because the set $S_{!e}$ is overestimated, finding $\Box S_{!e} = \emptyset$ proves that $S_{!e}$ has no solution and implies, with guarantee, that the hypothesis $\sigma \leq \epsilon$ is verified for every model entries.

On the contrary, finding a solution in $\Box S_{!e}$ does not prove that $S_{!e} \neq \emptyset$, due to the overestimation. In practice, it suggests that the hypothesis is probably false for one or more entries, but there is no theoretical guarantee. That is why this approach is used by a practitioner in the hope of obtaining that $S_{!e} = \emptyset$ for given entries.

5.2 Quantifying the error with IA

The interval methods introduced in Section 4.2.2, which deal with constrained optimization, can be used to provide bounds of variables appearing in the definition of S_e. In practice, it consists in adding to the constraint system (4) a goal to be optimized. Consider a given parameter $\rho \in \Phi$ (several ones can be handled one by one) The minimal parameter value satisfying the hypothesis, expressed by $S_\rho = Min(\rho), \rho \in \Phi$ such that $S_e = \Sigma$ is therefore obtained with the constrained optimization system:

$$\begin{cases} Minimize\ \rho\ s.t.: \\ M(e,\rho) = \nu \\ Dist(\nu - M_h(e_h,\rho_h)) \leq \epsilon \end{cases} \tag{10}$$

The algorithm used permits to find the minimal ρ which certifies that $S_{!e} = \emptyset, \forall e \in \Sigma$.

In the same manner, it is possible to find the maximal error σ by modifying the constraint defining the sub-set S_e:

$$\begin{cases} Maximize\ \sigma\ s.t.: \\ M(e,\rho) = \nu \\ \sigma = Dist(\nu - M_h(e_h,\rho_h)) \end{cases} \tag{11}$$

6 Application in robotics

In order to illustrate our approach, we apply it to our problem. We are interested in cable-driven robots within the context of a national granted project named CoGiRo. The final goal of this project involving researchers, engineers and PhD students, is to build a giant crane based on a parallel cable-driven robot. This raises numerous issues: design, mechanical conception, modeling, vision-based control, etc.

In this section, we introduce cable-driven robots, the cable model and finally highlight considerations that are specific to the control by cables (different from classical rigid actuators). We will focus on the inverse kinematics, a static is only used for the cable model (not for the robot equilibrium). We do not deal with dynamic model.

6.1 Cable-driven robots

A parallel cable-driven robot is made of a mobile platform (end-effector) connected to a fixed base by m cables. These cables can vary in length by the actuation of m pulleys linked to m rotary engines. The variation in length and tension of cables generates a movement in n degrees of freedom (position and/or orientation).

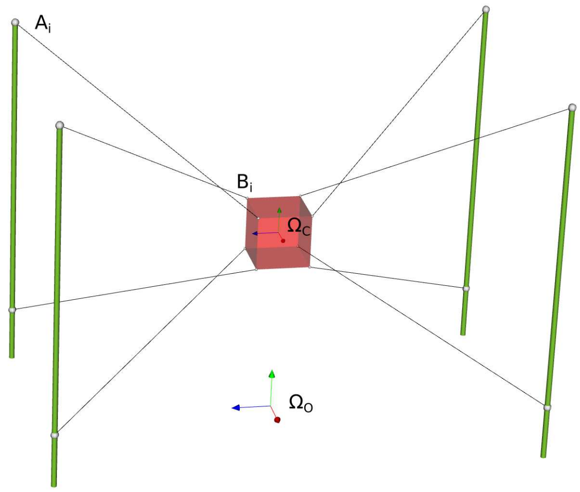

Figure 1: A cable-driven robot example

In the example presented in Fig. 1, the mobile platform or end-effector (mobile reference frame Ω_C) is connected to the base (fixed reference frame Ω_O) by $m = 8$ cables ($m > n$ to be fully controllable [9]). The i^{th} cable connects the point A_i of the base (coordinate a_i in Ω_O) to the point B_i on the mobile platform (coordinate b_i in Ω_C). The pose of the mobile $X = (P, R)$ (defined by the position P and the orientation matrix R of Ω_C w.r.t. Ω_O) is directly controlled by the length and the tension in each cable.

The workspace W_X is the set of all possible couples (P, R) for the robot.

6.2 Cable model

Cable-driven robots take advantage of the use of cables, allowing large workspace, light actuators in comparison to the possible load mass, and low cost. However, cable-driven robots suffer from the complex kinematics and dynamics of cables.

A well-known realistic model that is often used for the kinematics of cables is proposed by Irvine in [4]. In the Irvine model, the length of a cable depends on its tension. It is given for one cable and the equations are expressed in a plane made of the points $\overline{A}$ and $\overline{B}$ and the gravitational force:

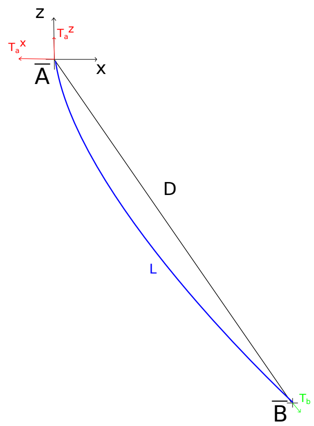

Figure 2: A cable in a plane $\{\overline{A}, \overline{B}, \overrightarrow{P}\}$

The Irvine model considers the geometric and static parameters of the configuration and the cable properties:

- Attachment points $\overline{A}$ (on base) and $\overline{B}$ (on platform)
- Cable: linear mass m, tightness (stiffness) k and length L
- Applied Tensions: T_b in $\overline{B}$ and T_a in $\overline{A}$

The system of three equations to be solved in order to obtain the actual length of cable and the tension distribution on point $\overline{A} = [0, 0]$ is:

$$\begin{cases} \overline{B}_x = & \frac{T_a^x L}{k} + \frac{|T_a^x|}{mg}[sinh^{-1}(\frac{T_a^z}{T_a^x}) - sinh^{-1}(\frac{T_a^z - mgL}{T_a^x})] \\ \overline{B}_z = & \frac{mgL^2}{k}(\frac{T_a^z}{mgL} - \frac{1}{2}) + \frac{1}{mg}[\sqrt{T_a^{x\,2} + T_a^{z\,2}} - \sqrt{T_a^{x\,2} + (T_a^z - mgL)^2}] \\ T_b = & \sqrt{T_a^{x\,2} + (T_a^z - mgL)^2} \end{cases} \quad (12)$$

This non-linear system is often solved numerically in T_a^x, T_a^z and L in function of the other parameters.

6.3 Consequences

The fact that a robot is controlled with cables, which have complex kinematics, leads to some problems in the classical fields of robotics:

- complex control;
- unworkable existing methods for the workspace determination;
- complex design;
- unsolvable modeling;

- unfeasible self-calibration, although a rough calibration using external measurement remains possible.

The hypothesis of non-elastic and mass-less cables is very useful to simplify control, modeling, calibration, etc. Moreover, this hypothesis is often realistic and generate a negligible error in robot accuracy.

The majority of papers dealing with these subjects use the hypothesis of mass-less and non-elasticity of cables, and replace the real length of cables L_i (depending on tensions) by the distances $D_i = A_iB_i$, $i = 1..m$. Under this assumption, the model is highly simplified with $L_i = D_i$.

The error between the real length and the distance AB needs to be quantified in order to check whether it remains below the accuracy of the actuators. This is the main purpose of the work described in this paper.

6.4 Problematics

We have seen above that the hypothesis of non-elasticity and mass-less done on cables properties is required (and often implicitly done) to hope to succeed in one of the major robotic fields applied to cable-driven robots. In our research, we have done this hypothesis in order to study the inverse kinematics of a robot similar to the one presented in Section 6.1. The hypothesis has to be checked on one cable before any static or dynamic modeling which are currently not mastered by the community. Nevertheless, note that our approach can be used for a more complex model with static or dynamic consideration.

6.5 Checking of non-elasticity and mass-less hypothesis

Our problematics is therefore to verify this hypothesis in the whole workspace of the robot W_X to bring the guarantee that the simplification is valid.

For this purpose, we compute the errors $\sigma_i = |L_i - D_i|$ made between the length L_i given by Irvine's model -function of B_i, T_{B_i} and the cable parameters- and the distance D_i, only function of B_i. B_i itself function of $X \in W_X$.

We then verify that these errors all lie under an acceptable threshold ϵ (which could be selected in function of the expected articular accuracy): $\sigma_i \leq \epsilon$, $i = 1..m$.

The position of the m points B_i are function of the pose $X = (P, R)$: the coordinates of B_i in Ω_O are $e_i = P + R.b_i$, b_i being the coordinates of B_i expressed in the platform reference Ω_C (defined by the platform geometry).

The hypothesis is verified on a pose $X \in W_X$, if for the every m points e_i: $\sigma_i \leq \epsilon$.

Therefore, we can define the subset S_X of acceptable poses as follows:
$S_X = \{X \in W_X, \forall i \in 1..m : \sigma_i \leq \epsilon\}$.

The proposed verification consists in checking the hypothesis in all the poses of the workspace. A sufficient condition is based on the dual set $S_{!X} = \{X \in W_X, \exists i \in 1..m : \sigma_i > \epsilon\}$. Interval methods can determine if $S_{!X} = \emptyset$, which implies the hypothesis holds on W_X. We remark that $S_X \cup S_{!X} \equiv W_X$.

Moreover, the m points B_i depending on X all belong to the same parallelepiped, whatever can be $X \in W_X$. Thus, we are satisfied with testing the hypothesis for only one cable. In addition, the parallelepiped built with the e_i is entirely covered by the diagonal plane with a simple rotation around the z axis.

Overall, the study of the workspace W_X can be reduced to the one of the diagonal plane W_B, as shown in Fig. 3. The point $B \in W_B$ could be expressed in the plane reference frame like in Irvine's model definition: $B = [\overline{B}_x, \overline{B}_z]$.

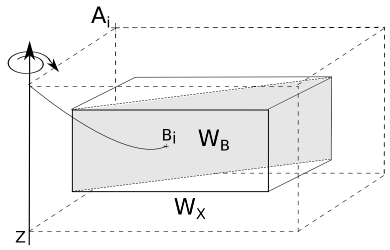

Figure 3: Robot space frame in dashed lines, workspace W_X and diagonal plane W_B

With this simplification, we define the sub-space of point B where the hypothesis is valid: $S_B = \{B \in W_B : \sigma \leq \epsilon\}$

And we will introduce the complementary: $S_{!B} = \{B \in W_B : \sigma > \epsilon\}$.
By construction of the simplification, if a solution is found in $S_{!B}$, a solution exists in $S_{!X}$ and the hypothesis is not valid in the whole workspace.
In the same manner, we remark that $S_B \cup S_{!B} \equiv W_B$
It's also easier to find zero solution in $S_{!B}$ that prove that $S_B \equiv W_B$. Proving that $S_{!B}$ has no solution implies indeed that the hypothesis $\sigma \leq \epsilon$ is verified for every point in the workspace.

6.6 Quantifying the error

The sets S_B and $S_{!B}$ give an answer about the validity of the hypothesis. However, we could expect additional quantified information such as:

- The minimal tension satisfying the hypothesis, expressed by $S_{T_b} = Min_{T_b}$, $\forall B \in S_B$
- The maximal error committed in the workspace, defined by $S_\sigma = Max_\sigma$, $\forall B \in W_B, \forall T_b \in [T_{min}, T_{max}]$

6.7 Interval strategy and problem adaptation

The constraint system is based on Irvine's model. We do not use directly the system of three equations presented in Section 12; we prefer a system with 5 additional variables and equations that allows a faster solving process. This system manipulation is performed in order to eliminate the division and to replace $sinh^{-1}$ by its logarithmic expression $sinh^{-1}(x) = ln(x + \sqrt{(x^2+1)})$. The constraint system to compute a superset of $S_{!B}$ (every real-valued element being approximated by interval) is:

$$\begin{cases} mg\overline{B}_x - eT_a^x + T_a^x \ln\left((h-c)(c-d+j)\right) = & 0 \\ mg\overline{B}_z - T_a^x h - T_a^z e + \dfrac{ke^2}{2} + T_b = & 0 \\ T_b - T_a^x j = & 0 \\ cT_a^x - T_a^z = & 0 \\ h - \sqrt{(c^2)+1} = & 0 \\ j - \sqrt{((c-d)^2)+1} = & 0 \\ dT_a^x - mgL = & 0 \\ ek - mgL = & 0 \\ |L - \sqrt{\overline{B}_x{}^2 + \overline{B}_z{}^2}| > & \epsilon \end{cases} \tag{13}$$

This model is not a simplification of 12, but a rewriting. These models are absolutely equivalent. A solution of this constraint system is provided by an 11-dimension box defined by: $[\overline{B}_x] \times [\overline{B}_z] \times [T_a^x] \times [T_a^z] \times [T_b] \times [L] \times [c] \times [d] \times [e] \times [h] \times [j]$. Only the projection onto $[\overline{B}_x] \times [\overline{B}_z]$ is interesting for us to find a point B in $\square S_{!B}$.

7 Experimental results

7.1 Our robot

The prototype, shown in Fig. 4, was built by the TECNALIA company (www.tecnalia.com) in collaboration with the LIRMM laboratory (www.lirmm.fr).

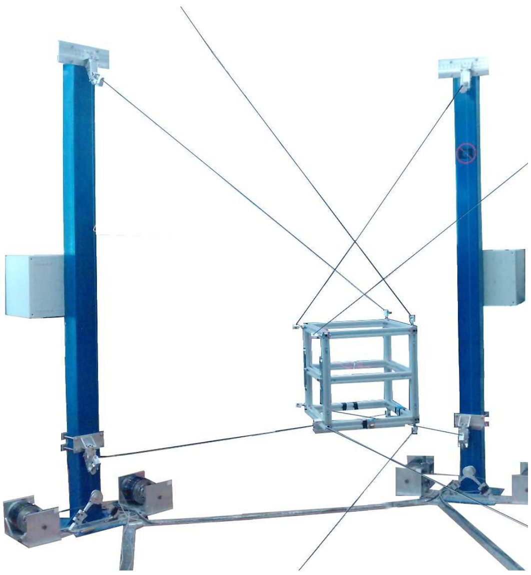

Figure 4: Partial view of the prototype ReelAx8

ReelAx8 is a reconfigurable cable driven robot. Eight cables, wound on winches, are attached by spherical joints to the eight corners of a cube shaped platform of about 40 centimeters large. Four pairs of winches are fixed on posts up to three meters arranged at the four corners of a 3 meters by 4 meters rectangle.

The prototype is given with a rectangular workspace of 2 meters (from 1 to 3 meters on x axis) by 1 meter (from 1 to 2 meters on y axis) on floor, 1 meter high (from 1 to 2 meters on z axis) and ± 5 degrees of rotation on each axis.

We can use the notation: $W_X = \{X = [P, R], P \in [1,3] \times [1,2] \times [1,2], R \in [-5,5] \times [-5,5] \times [-5,5]\}$

7.2 Experiments

The solving is performed by adapting an interval tool developed in the COPRIN team and briefly described in Sections 4.2 and 4.3. It is made with a contractor using monotonicity and shaving, and a bisector using the derivative of constraints.

In our process, we uncorrelate the lower tensions case which leads to the supremacy of the cable curvature over the elasticity ($\sigma > 0$) and the higher tensions case which results in the inverse preponderance ($\sigma < 0$).

7.2.1 Case 1: existing prototype

The cables used have the following characteristics: $k = 137kN/m$, $m = 0.007kg/m$.
With the workspace W_X introduced in the description of the robot, the plane W_B to be tested is the diagonal plane of the rectangular parallelepiped $[1,3] \times [1,2] \times [1,2]$: $W_B = [1, 3.7] \times [1,2]$.
We fix $\epsilon = 0.005m \simeq$ expected accuracy of robot.

The sensors give, during our tests, a minimal tension of $20N$ and a maximal one of $120N$. So $20 \leq T_b \leq 120$.

7.2.2 Case 2: robot under construction

We consider the same architecture robot but with heavier cables and larger workspace. The cables are in the same steel with a tightness $k = 137kN/m$, and a lineic mass $m = 0.092kg/m$. The workspace is estimated at $W_B = [1,8]*[1,10]$ for the next prototype for which we also expect an accuracy of 1 cm. The tension should be between $40N$ (for a just tightened cable) and $1000N$ (at maximal load).

7.3 Results in term of performances

In this paragraph, we present the results of our different strategies in term of performances, time and number of boxes created during the interval-based solving process. These different tests lead us to adjust our strategy for the remaining tests. They are performed for the evaluation of $S_{!B} = \emptyset$. Indeed, for the evaluation of $S_{!B} \neq \emptyset$, i.e when the hypothesis is non acceptable, a solution is found quickly, which does not allow to compare the strategies.

The first results in Table 1 are given for different bisectors and for the model (12) and the modified model (13). They show that the rewritten system is greatly better for our solving process both in term of time elapsed and in term of number of boxes created. It is also more sensitive to the Smear bisector, when the usual system yield to worst performances with this strategy choice. This phenomenon comes from the instability of the usual system.

	System (12)		System (13)	
Bisector	Time(s)	Boxes(10^6)	Time(s)	Boxes(10^6)
RR	18173	14	5772	2
Smear	18369	14	5120	2

Table 1: Choice of system and bisector (with contractor 3B(HC4) and W_B=[1,4]*[1,3])

Contractor	Slices	Time(s)	Boxes
3B(HC4)	10	timeout	/
3BCID(HC4)	10	7790	107000
3BCID(HC4)	1000	1439	52157
3BCID(HC4)	10000	720	3729
3BCID(Mohc)	1000	841	15299
3BCID(Mohc)	10000	547	2199

Table 2: Choice of contractor (with Smear, system (13) and W_B=[0.5,4]*[0.5,3])

The results presented in Table 2 come from experiments on the choice of the contractor. They show that the performance increases with the number of slices. However, contrarily to bisection, the number of slices of 3B or 3BCID are achieved on only one dimension at a time (although the slices must be small). The reason for which it is efficient on this problem could be the small number of variables in the constraints of system (13). Indeed, this tends to increase the power of constraint propagation (HC4 or Mohc) and thus the chance of eliminating a given slice.

To conclude on the research of the best strategy, we definitely select the 3BCID(Mohc) with 10000 slices and confirm the rewriting of Irvine's model in order to improve the efficiency of our algorithm (particularly for the Smear add-on). Consequently, in the following experiments, we will use this strategy to obtain the best efficiency.

To illustrate the contractor and bisector processing, the subpaving is drawn in Fig. 5.

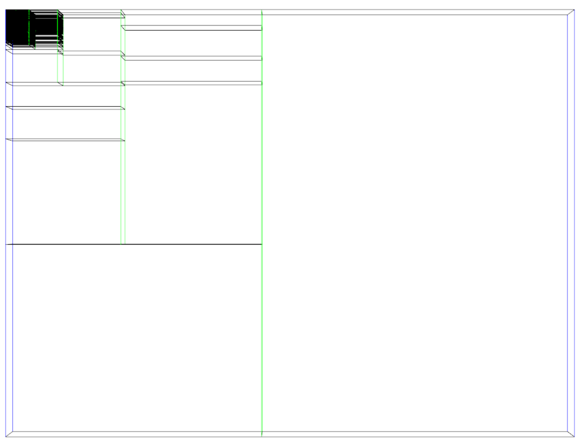

Figure 5: Rejected boxes during process

This illustration provides information about the regions easy to eliminate or contract (large boxes) and the more difficult regions (small boxes) where the system is very unstable. The difficult region is close to the singularity of the cable model (i.e., close to the vertical plane/line).

7.4 Hypothesis confirmation

We compute an overestimation of the subset $S_{!B}$, noted $\Box S_{!B}$, by using the constraint system (13). Recall that if no solution is found in $\Box S_{!B}$, no solution exists in $S_{!B}$, and the hypothesis is valid in the considered workspace.

7.4.1 Case 1

No solution is found by our tool, therefore the hypothesis is acceptable for the studied robot. The model using the simplification is thus sufficiently accurate. The solving process achieved in the whole workspace take about 2 hours. For a reduced workspace, for example one by one meter, the resolution is performed in about 10 minutes.

7.4.2 Case 2

A solution for $\Box S_{!B}$ is immediately found ($\approx$ 1 second). The hypothesis seems therefore too strong and a more complex model must be developed for the giant robot under-construction. Otherwise, the robot model accuracy could be highly deteriorated.

7.5 Global optimum searching

In addition to the yes/no results obtained about the hypothesis validation, global optimization gives the opportunity to enrich the knowledge about robots. First, our method can provide the maximal error committed in the workspace and defined previously by:
$S_\sigma = Max_\sigma, \forall B \in W_B, \forall T_b \in [T_{min}, T_{max}]$.

Second, it can also compute the minimal tension, satisfying the hypothesis, expressed by:
$S_{T_b} = Min_{T_b}, \forall B \in S_B$.

More generally, we will see that it is possible and often easy to find different optima which better define the design of the robot and the quality of the kinematic model.

7.5.1 Case 1

The analysis of the prototype model (for which the hypothesis has been proved acceptable by our verification method) provides useful information gathered in Table 3

- Minimal T_b to keep $|\sigma| < \epsilon$, see Table3, column 1;
- Maximal $|\sigma|$ for $T_b = 20N$, see Table3, column 2;
- Maximal $|\sigma|$ for $T_b = 120N$, see Table3, column 3;
- Maximal T_b to keep $|\sigma| < \epsilon$, see Table3, column 4;

The values found confirm the hypothesis validation.

	1	2	3	4
Error σ (m)	0.005	**0.0006**	**-0.0037**	-0.005
Tension T_b (N)	**1.4**	20	120	**171.5**
Time (s)	200	3	5	30

Table 3: Results (in bold) obtained by optimization processes on the existing prototype (case 1)

	1	2	3	4
Error σ (m)	**0.01009**	0.01	-0.01	**-0.093**
Tension T_b (N)	40	**40.1**	**121**	1000
Time (s)	72	3	6000	5

Table 4: Results (in bold) obtained by optimization processes on the robot under construction (case 2)

7.5.2 Case 2

The same model analysis protocol is followed for the cable-driven robot under construction and the results are presented in Table 4.

- Maximal $|\sigma|$ for $T_b = 40N$, see Table4, column 1;
- Minimal T_b to keep $|\sigma| < \epsilon$, see Table4, column 2;
- Maximal T_b to keep $|\sigma| < \epsilon$, see Table4, column 3;
- Maximal $|\sigma|$ for $T_b = 1000N$, see Table4, column 4;

The values found confirm the hypothesis rejection, even if the lower tension bound is close to the minimal tension for which the hypothesis is valid.

7.6 Conclusion on experiments

The application chosen to demonstrate the process is related to the main task of the author: the calibration of parallel cable-driven robots. In our research, we have done the hypothesis of mass-less and non-elasticity of cables in order to self-calibrate the robot presented in Section 7. Indeed, to self-calibrate a cable-driven robot, we must consider it as a redundantly actuated manipulator. This redundancy is conditioned by the independence of cables length from their tension. This condition is obtained with the simplified model under non-elasticity and mass-less assumption. This hypothesis is validated with our method, the simplification is thus acceptable and this robot is self-calibratable. In the second case, the robot under construction, the hypothesis is rejected. To self-calibrate this giant crane, we must find a sub-workspace where the hypothesis is acceptable. Other interval methods build so-called *inner* boxes, i.e., boxes in which all points are solutions. For the second robot, it could be interesting to pave the workspace with numerous inner boxes, thus finely defining the zone of the workspace where the hypothesis is respected. Even further (and more costly), one could pave the entire workspace with several sets of boxes, each set containing boxes with the same error range.

To conclude, we have designed an operational tool for analyzing the difference between a real cable model and a strong simplification of it. The method described in this paper has provided interesting and useful results for our study of cable-driven robots. Its implementation represents a first software version of a dedicated design tool which could be incorporated in an "Appropriate design" approach [8].

Finally, this dedicated tool can be useful for modeling, designing and optimizing in a reliable way robots, but also other mechanisms that make use of cables.

8 Discussion and conclusions

We propose in this paper a quasi-generic method to confirm or reject an hypothesis used to simplify a model. This checking is done by analyzing the difference between a model and its simplified version. Our approach based on interval analysis allows to give more information on the simplified model such as the maximal error done in a whole use model field, or parameter bounds to keep the simplified model close to the realistic model.

Numerous experiments have been performed to illustrate our approach and to justify the choice of the strategy and of the constraint system form. Moreover, the tool developed for these experiments is useful for the design or the kinematic studies in the cable-driven robot field.

Our approach could therefore provide many tools for qualifying a simplified model in different fields like mechanics, chemistry, biology.

Acknowledgment

This work was supported in part by the French National Research Agency (ANR) under grant 2009-SEGI-018 (CoGiRo ANR project) and in part by the Région Languedoc-Roussillon under grant 115217.

References

[1] I. Araya, G. Trombettoni, and B. Neveu. Exploiting Monotonicity in Interval Constraint Propagation. In Proc. AAAI, pages 9–14, 2010.

[2] E. R Hansen. Global Optimization Using Interval Analysis. Marcel Dekker Inc., 2003.

[3] R. Horst and H. Tuy. Global Optimization: Deterministic Approaches. Springer, 1996.

[4] H. M. Irvine. Cable Structures. MIT Press, 1981.

[5] L. Jaulin, M. Kieffer, O. Didrit, and E. Walter. Applied Interval Analysis. Springer, 2001.

[6] R.B. Kearfott and M. Novoa III. INTBIS, a portable interval Newton/Bisection package. ACM Trans. on Mathematical Software, 16(2):152–157, 1990.

[7] O. Lhomme. Consistency Techniques for Numeric CSPs. In IJCAI, pages 232–238, 1993.

[8] J-P. Merlet and D. Daney. Appropriate design of parallel manipulators. Smart Devices and machines for advanced manufacturing, pages 1–25, 2008.

[9] A Ming and T Higuchi. Study On Multiple Degree-Of-Freedom Positioning Mechanism Using Wires .2. Development Of A Planar Completely Restrained Positioning Mechanism. International Journal Of The Japan Society For Precision Engineering, 28(3):235–242, Sep 1994.

[10] R. E Moore. Interval Analysis. Prentice-Hall, 1966.

[11] A. Neumaier. Interval Methods for Systems of Equations. Cambridge Univ. Press, 1990.

[12] A. Pacut and W.J. Kolodziej. Validity of model simplification. In Decision and Control, 1990., Proceedings of the 29th IEEE Conference on, pages 2904 –2905 vol.5, dec 1990.

[13] G. Trombettoni, I. Araya, B. Neveu, and G. Chabert. Inner regions and interval linearization for global optimization. In AAAI 2011, San Francisco, CA, USA, 2011.

[14] G. Trombettoni and G. Chabert. Constructive Interval Disjunction. In Proc. CP, LNCS 4741, pages 635–650, 2007.

RESEARCH CENTRE
SOPHIA ANTIPOLIS – MÉDITERRANÉE

2004 route des Lucioles - BP 93
06902 Sophia Antipolis Cedex

Publisher
Inria
Domaine de Voluceau - Rocquencourt
BP 105 - 78153 Le Chesnay Cedex
inria.fr

ISSN 0249-6399