INTERNATIONAL DESIGN CONFERENCE - DESIGN 2004
Dubrovnik, May 18 - 21, 2004.

DESIGN 2004

TRUSS DIMENSIONING WITH AN UNCERTAINTY REDUCTION PARADIGM

B. Yannou and A. Hamdi

Keywords: constraint programming, preliminary design, management of uncertainty, design space, interval analysis

1. Introduction

The management (representation and propagation) of uncertainty is known to be of the utmost importance in the preliminary design stages of product design (see [Antonsson *et al* 1995; Yannou 2003]). Indeed, the uncertainty reduction paradigm turns out to be much more virtuous for concurrent engineering in comparison to the try-and-test deterministic optimization paradigm [Ward *et al* 1994]. Unfortunately, few uncertainty management systems exist for they face a number of severe limitations such as: the size of the problems, the computation times, the lack of consistency between the "uncertain" representations of the design variables. In previous papers [Yannou 2003; Yannou *et al* 2003], we presented the outlines of the three families of methods for managing uncertainty, namely: fuzzy methods [Antonsson *et al* 1995], probabilistic methods [Thurston *et al* 1991] and Constraint Programming (CP) methods where the variables are represented by their domains (discrete or continuous, i.e. intervals). We advocated that *CP techniques over reals* advantageously compete with the two others families of methods for an use in preliminary design (see for example [Lottaz *et al* 2000; Merlet 2001]) or, at least, that they worth to be further studied because of the number of good properties and to recent significant advances.
After a brief evocation, in chapter 2, of the principles and properties of CP techniques over reals, this article details the scenario of an interactive preliminary dimensioning of a truss structure. The design problem is presented in chapter 3 and its modelling in a CP context in chapter 4. The peculiar flexibility for specifying new requirements on the design, for assessing the consequences of a given design action on the shape of the remaining design space and for better understanding the nature of relations between design parameters and performances variables are highlighted through a design scenario in chapter 5 before concluding.

2. Brief overview of constraint programming over reals

With *Constraint Programming* (CP) *over reals*, variables are modelled as intervals of allowable values and CP techniques are sophisticated evolutions of interval analysis or *interval arithmetics* (see [Moore 1979]). Starting from a set of initial domains for the imprecise variables and from a set of mathematical constraints linking the variables, different CP *consistency techniques* (such as *Hull*, *Box*, *weak-3B* or *3B*, see [Benhamou *et al* 1999; Granvilliers 2002; Granvilliers *et al* 2001]) try to contract at most the variable domains so as to eliminate unfeasible values. One tries to result in the more tightened Cartesian product of intervals, ensuring at any moment that all feasible solution is kept inside (it refers to the *completeness* property). When the mechanism of domain bisection is recursively applied in parallel with the contraction mechanism, a research tree is build until a stopping criterion (width of the domains or number of enumerations) is reached. This branch-and-prune algorithm allows

to prune out large parts of the design space whenever a domain is proved to be empty. At the end, the design space is approximated by a number of elementary Cartesian products of small intervals, denoted *boxes* in the following. The hull of these boxes provides a valuable information to the designer concerning the potential values remaining for any variable at this stage. Lastly, a graphical representation of this collection of *n*-dimensional boxes (*n* being the number of constrained variables) is easy and convenient to get some good pictures of the design space. The design space can be apprehended by its two or three-dimensional projections on couples or triplets of variables (see [Sam 1995]). Here, we have used a representation tool named *Universal Solution Viewer* (*USV*) [Christie 2002].
We believe that modelling and solving stages of a design problem with CP techniques are *appropriate* but require a *particular tuning* in CP modelling and solving. They are appropriate because :

- a design problem is characterized by significant uncertainties on variable values (large variable domains) at the beginning of a dimensioning process and the designers are wishful to benefit from a precise and consistent representation of the remaining design space at any moment,
- the constraints linking design variables are often of an heavily intricate polynomial form and CP techniques are efficient techniques for solving non-linear constraints,
- the completeness property that guaranties to keep any feasible design solution in the approximate design space is a crucial property in design,
- the consistency of CP techniques (efficiency in contracting the domains) is much better than with fuzzy techniques (see [Dong et al 1987; Wood et al 1989]),
- the uncertainty reduction is naturally propagated in both directions: from design parameters to performance variables and conversely,
- graphical explorations of the design space are natively easy.

A fine description of CP consistency techniques, of the branch and prune algorithm and of their tuning are beyond the scope of this paper. Let us simply mention that we have used the CP platform, named RealPaver (see [Granvilliers 2002]), developed by the IRIN computer science department of the Nantes (France) university. We will refer to some other properties of CP techniques when necessary throughout the design study of the truss structure.

3. The truss structure design problem

Our case study consists in dimensioning the two members of a truss structure described in figure 1. This problem has been originally proposed by Wood et al [Wood *et al* 1989] to compute imprecise performance parameters from imprecise design parameters via fuzzy techniques. This example has also been used by Scott et al [Scott *et al* 2000] in a different parameterized form to select an optimal Pareto solution that could not be selected via a linear aggregation function (importance weights).
In the following, we have exactly adopted the parameterization and the initial variable domains of the truss structure of Wood et al [Wood *et al* 1989] but we have adopted the more complex design constraints and performances of Scott et al [Scott *et al* 2000].

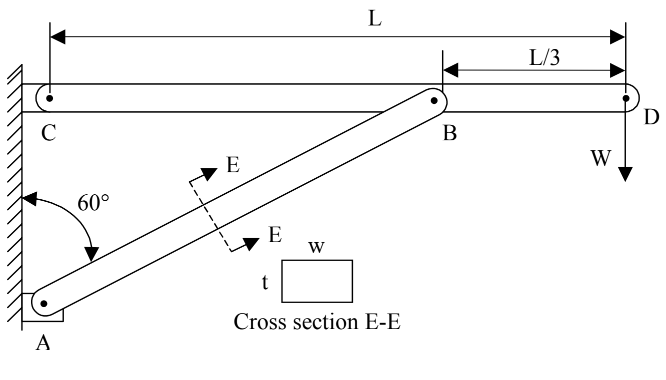


Figure 1. The parameterization of the truss structure

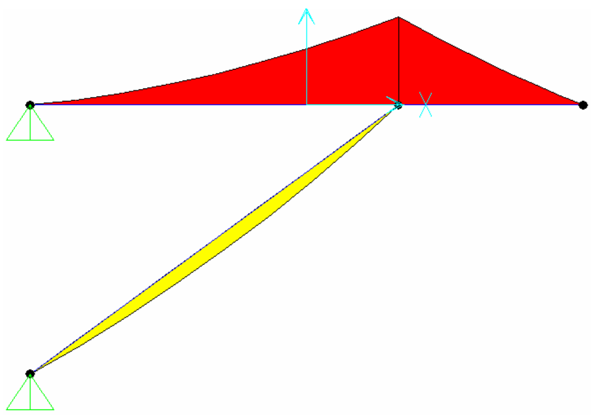


Figure 2. The bending momentum

The need is to design a mechanical structure supporting an overhanging vertical load at a distance L from the wall with a minimal mass. One possible configuration (see figure 1) consists in a two-member pin-jointed bracket with an horizontal member (CD) and a compression member (AB) attached to the wall at an angle of sixty degrees. The common pin is located at two thirds of L from the wall. Both members have rectangular cross sections: $w_{AB} \times t$ for (AB) and $w_{CD} \times t$ for (CD), w standing for width and t for thickness. Additional design choices have been made: The material of both members is steel and we impose $w_{CD}=w_{AB}-0.025$. The designer has to make decisions on values of the following *design variables*: t, w_{AB} and L. Moreover, the specification on the overhanging load W is imprecise (between 15 and 20 kN); in consequence, W is considered as a fourth design variable.
The two *mechanical constraints* to respect are the followings:

- the maximal bending stress σb in member (CD) must be lower or equal to the allowable bending limit σr (here, 225 MPa for steel).
- the compression force FAB in member (AB) must be lower or equal to the buckling limit Fb.

The maximal bending stress σ_b is located at point B (see the bending momentum in figure 2) and is given by the following formulas involving W_{CD}, the weight of member (CD):

$$\sigma_b = \frac{2L\left(W + \frac{W_{CD}}{6}\right)}{w_{CD}t^2} \quad \text{with} \quad W_{CD} = \rho g w_{CD} t L \quad \text{and} \quad w_{CD} = w_{AB} - 0.025 \tag{1}$$

The compression force in member (AB) is given by the following formulas involving W_{AB}, the weight of member (AB):

$$F_{AB} = \sqrt{\left\{\frac{9}{2\sqrt{3}}\left(W + \frac{W_{CD}}{2} + \frac{W_{AB}}{3}\right)\right\}^2 + \left\{\frac{3}{2}\left(W + \frac{W_{CD}}{2}\right)\right\}^2}$$

$$\text{with} \quad W_{AB} = \rho g w_{AB} t L_{AB} \quad \text{and} \quad L_{AB} = \frac{4\sqrt{3}}{9}L \tag{2}$$

Furthermore, the buckling limit in member (AB) is given by:

$$F_b = \frac{\pi^2 E I_{AB}}{L_{AB}^{\ 2}} = \frac{9\pi^2 E w_{AB} t^3}{64 L^2} \tag{3}$$

The *performance variables* are the mass M of the structure (to minimize) and the safety factor s, i.e. the factor of over-dimensioning beyond the strict respect of the two aforementioned constraints. The mass M is given by:

$$M = W_{AB} + W_{CD} \quad (4)$$

The safety factor of the truss structure s is the minimum between the safety factor below the allowable bending limit σ_b, namely s_σ, and the safety factor below the buckling limit F_b, namely s_F. They are expressed by:

$$s_\sigma = \frac{\sigma_r}{\sigma_b}, \quad s_F = \frac{F_b}{F_{AB}}, \quad s = \min(s_\sigma, s_F) \quad (5)$$

The two mechanical constraints may be merely expressed by: $s_\sigma \geq 1, \quad s_F \geq 1$ (6)

4. Modelling the constraint programming problem

For setting a constraint programming problem, the initial domains of the design and performance variables must be defined (see table 1). The initial domains of the design variables are those of Wood et al [Wood *et al* 1989]. In fixing the lower bound of the two elementary safety factors s_σ and s_F to 1, the mechanical constraints are taken into account. The lower bound of mass is just set to 0, with no further information.

Table 1. Initial domains for the design and performance variables

Design variables	Performance variables	Constants
$t \in [0.04, 0.10]$ $w_{AB} \in [0.04, 0.13]$ $L \in [3, 4]$ $W \in [15000, 20000]$	$M \in [0, +\infty[$ $s_\sigma \in [1, +\infty[$ $s_F \in [1, +\infty[$ $s \in [1, +\infty[$	$E = 207 \cdot 10^9\ Pa$ $\rho = 7830\ kg/m^3$ $g = 9.81\ m/s^2$ $\sigma_r = 225 \cdot 10^6\ Pa$

An important rule in the modelling of a CP problem is that intermediary variables the designers are not interested in must be eliminated from the set of constraints so as not to consider those variables in the design space and to get the best domain contraction for the actual design and performance variables. This is why:

- Variables WAB, WCD, wCD, Fb and LAB must be replaced by their expressions in t, wAB, L and W in the constraints.
- Variables sσ and sF must not be considered as performance variables the designers are interested in and, then, must not be bisected in the splitting process. A special mechanism exists in RealPaver [Granvilliers 2002] for hiding such variables in the enumerated boxes. It consists, when these variables are defined as functions of other variables, in internally symbolically replacing their occurrences by their expressions, although initial domains may be defined on them. These intermediary variables to hide are preceded by a $ sign in eq. (7).
- In addition, the number of occurrences of the same variables must be lowered as much as possible to avoid the dependency problem. The dependency problem (see [Granvilliers et al 2001]) is generated by the fact that a variable occurrence is brutally replaced by its current domain during the solving process. Subsequently, the multiple occurrences of a given variable within a given constraint and even between different constraints are decorrelated. This decorrelation results in relaxed constraints and then in larger domains. This is why it is often necessary to reformulate constraints in decreasing the numbers of the same variable occurrences by appropriate factorization strategies (see [Ceberio et al 2000]). The choice of a global consistency technique (to oppose to local consistency techniques like hull or box) like

the weak-3B-consistency here also partly overcomes this problem1 (see [Lhomme et al 1996]).

Finally, the constrained problem is entirely expressed by the four following constraints:

$$\begin{cases} \$s_\sigma = \dfrac{\sigma_r \left(w_{AB} - 0.025\right)t^2}{2L\left(W + \frac{1}{6}\rho g t L\left(w_{AB} - 0.025\right)\right)} \\ \$s_F = \dfrac{\left(\dfrac{3\pi^2 E w_{AB} t^3}{32 L^2}\right)}{\sqrt{3\left(W + \frac{1}{2}\rho g t L\left(w_{AB}\left(1 + \frac{8}{9\sqrt{3}}\right) - 0.025\right)\right)^2 + \left(W + \frac{1}{2}\rho g t L\left(w_{AB} - 0.025\right)\right)^2}} \\ M = \rho g t L\left(w_{AB}\left(1 + \frac{4\sqrt{3}}{9}\right) - 0.025\right) \\ s = \min\left(\$s_\sigma, \$s_F\right) \end{cases} \quad (7)$$

5. The design scenario

We have simulated a realistic design scenario in a constraint programming environment so as to stress the interactive nature of communication between the designer and its emerging design, i.e. the enrichment the CP analysis brings into the understanding of the design concept potentialities and even of the design objectives themselves (or of their prioritisation). A unique dimensioning of the truss structure has emerged after a four-steps design process. Each step has consisted in an enumeration of 1000 boxes of design space, using a *weak-3B-consistency* technique.

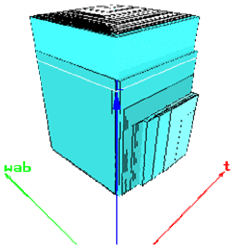
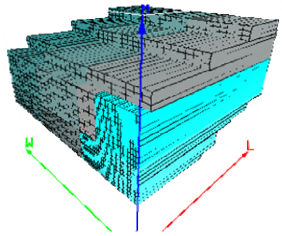
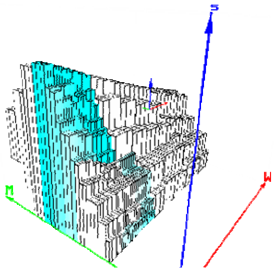
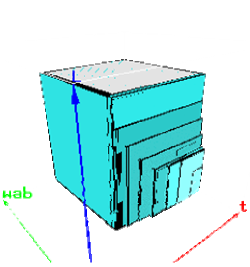
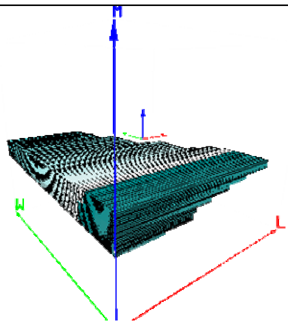
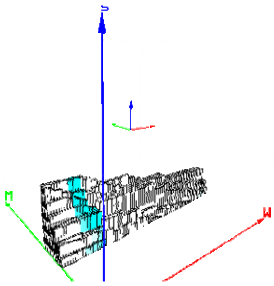
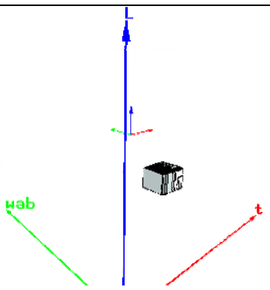
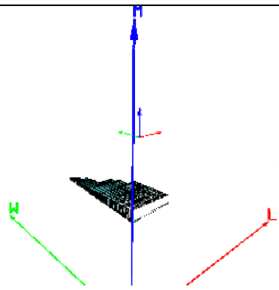
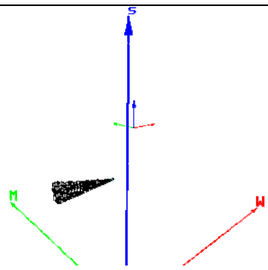
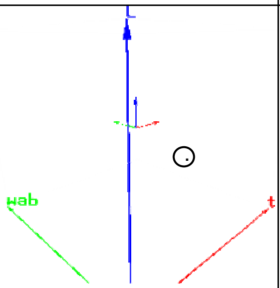
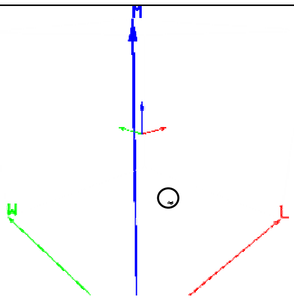
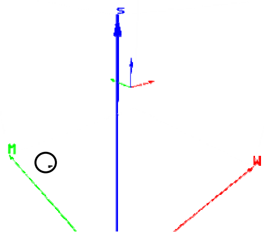
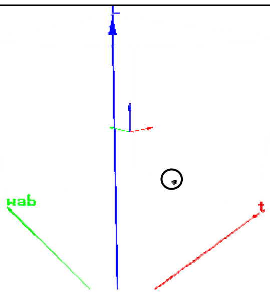
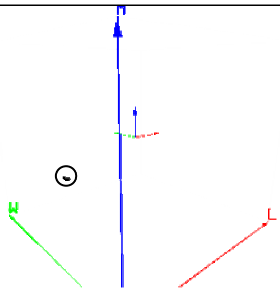
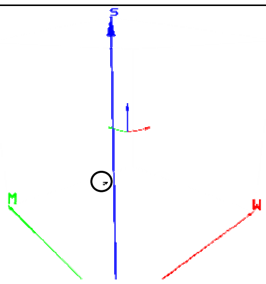
The **first design stage** is defined by table 1 and equations (7). The hull of the design space surrounding the enumerated boxes is calculated, already leading to sensible domain contractions (see the last column of table 2). Three 3-dimensional projections of the design space are made on $\{t, w_{AB}, L\}$, $\{L, W, M\}$ and $\{W, M, s\}$ (see table 2). The shape of the design space confirms some intuitive trends: The greater the supported weight *W* or the structure length *L*, the greater the structure mass *M* and the lower the safety factor. But the fact that it seems to exist a given safety factor for which the mass *M* is the greatest is not so intuitive and even reveals that simultaneously minimizing *M* and maximizing *s* is a hard task (see [Scott *et al* 2000]). Among remarkable results brought by this CP computation, *M* is ensured to stand between 2077.9 and 6300.9 kg, the safety factor is already ensured to be lower than 2.567 and the lower bounds of *t* and w_{AB} have been tightened.

Starting from this first result, the designer decides in a **second stage** to guaranty that the structure mass is under a certain control in adding the following constraint $M \leq 3200$. After recomputation, the design space has been dramatically shrunk. As a first consequence, the length *L* is no longer able to be of 4 m but is limited to 3.8 m. The safety factor domain is again tightened and *s* is ensured to be lower than 1.664.

In a **third stage**, the designer decides that the safety factor must also be guaranted, given that a safety factor close to 1 is risky. Hence, he decides to impose the additional constraint $s \geq 1.5$. Consequently, all the (hull) domains are further contracted. For instance, there is almost no more degree of freedom on *L* (between 3 and 3.15) and *M* (between 2926.5 and 3200). The domains of w_{AB} and *W* are suddenly tightened. The designer is now informed that the overhanging load is no more authorized to be greater than 16638 kg. Then, the designer is compelled to wonder if this requirement which was kept flexible is still required.

[1] Not detailed further here.

Table 2. The four-stages design scenario of the truss structure

Stage s	$\{t, w_{AB}, L\}$ projection	$\{L, W, M\}$ projection	$\{W, M, s\}$ projection	Hull of design space
1				$t \in [0.0621, 0.1$ $w_{AB} \in [0.0654,$ $L \in [3, 4]$ $W \in [15000, 20$ $M \in [2077.9, 6$ $s \in [1, 2.567]$
2 $M \leq 32$				$t \in [0.0621, 0.$ $w_{AB} \in [0.0654$ $L \in [3, 3.8]$ $W \in [15000, 2($ $M \in [2081.5, 1$ $s \in [1, 1.664]$
3 $s \geq 1.5$				$t \in [0.0887, 0.1$ $w_{AB} \in [0.0859,$ $L \in [3, 3.150]$ $W \in [15000, 16$ $M \in [2926.5, 3$ $s \in [1.5, 1.664]$
4 $L \geq 3.1$				$t \in [0.0996, 0.$ $w_{AB} \in [0.0889$ $L \in [3.14, 3.14$ $W \in [15000, 15$ $M \in [3190.6, 1$ $s \in [1.5, 1.505]$
4' $W \geq 16$				$t \in [0.0990, 0.$ $w_{AB} \in [0.0920,$ $L \in [3, 3.013]$ $W \in [16500, 1($ $M \in [3176.9, 1$ $s \in [1.5, 1.512]$

One could imagine two situations for the designer to focus on an unique deterministic design in a **fourth stage**. In a first situation (numbered 4 in table 2), the designer does not pay attention to the shape of the design space resulting in stage 3. Indeed, a fallacious reasoning leads him to impose the largest value for length *L*, thinking that the larger the structure, the larger the overhanging load *W* and the larger the safety factor *s*. Thus, as the domain hull of *L* was $[3, 3.150]$ in stage 3, the designer has decided to impose the constraint $L \geq 3.14$ to stick *L* to its largest value. But the consequence is quite calamitous since *s* and *W* are stuck to their lower bound, i.e. respectively 1.5 and 15000 kg. In fact, it was a wrong analysis because both performances *s* and *M* were already strongly constrained and an increase of *L* probably tends to increase *M* and lower *s* by too much.

In a second situation (the virtuous one, numbered 4' in table 2), the designer prefers to go on specifying only requirements (objectives) and not design parameters (means or solutions) without presuming the link between them. Moreover, he studies in details the graphical results obtained in stage 3. Thus, he decides to stick the value of the overhanging load *W* to its upper bound with the additional constraint $W \geq 16500$ since he reminds that the requirement was fuzzy in the sense that *W* could be possibly greater than 15000 kg. Here a fuzzy preference (i.e. a value of possibility associated to each value of the variable domain) would have better captured this notion but one can see here that it can be done without. Here, contrarily to the first situation, *L* is stuck to its lower bound and, above all, *W* is stuck to its upper authorized bound, i.e. 16600 kg, for the same mass *M* and safety factor *s*. In addition, it can be noted that all the variables have been stuck to one of their bounds except for the member width w_{AB} that reaches a slightly different optimal value in stages 4 and 4'. Such a result is not trivial from the sole expression of constraints.

6. Conclusions

We believe that Constraint Programming over continuous domains are valuable and promising techniques to model and propagate uncertainties on the variable values during the conceptual design stage. They result in an encompassing design space that is easy to represent and which may bring a pertinent information for a personal or an inter-personal negotiation (see for instance [Lottaz *et al* 2000]). A CP environment is well suited to test *if-then* reasoning and to result in a better comprehension of the often complex coupling between design parameters and performance variables. This apprenticeship may be necessary to become conscious of the importance of relaxing some requirements on performances. The design scenario on the truss structure also revealed that, when the designer constrains the performances in a functional manner, its design action is automatically back-propagated towards the domains of the design variables, demonstrating a high-level synthesis capability.

References

Antonsson E.K., Otto K.N., "Imprecision in Engineering Design", Journal of Mechanical Design, Vol.117, No.B. 1995, pp. 25-32.

Benhamou F., Goualard F., Granvilliers L., Puget J.-F., "Revising Hull and Box Consistency", in Procs. of ICLP'99, The MIT Press, Las Cruces, USA, 1999.

Ceberio M., Granvilliers L., "Solving Nonlinear Systems by Constraint Inversion and Interval Arithmetic", AISC: 5th Int. Conf. on Artificial Intelligence and Symbolic Computation. Madrid, Spain, 2000, pp. 127-141.

Christie M., "USV: user manual", IRIN Lab, Nantes University, Nantes, France, 2002.

Dong W.M., Wong F.S., "Fuzzy weighted averages and implementation of the extension principle", Fuzzy Sets and Systems, 1987, pp. 183-199.

Granvilliers L., "RealPaver user's manual. Version 0.2, http://www.sciences.univ-nantes.fr/info/perso/permanents/granvil/realpaver/main.html", Univ. of Nantes, Lab of Computer Science, 2002.

Granvilliers L., Benhamou F., Huens E., "Constraint Propagation (chapter 5)", in COCONUT Deliverable D1 - Algorithms for Solving Nonlinear Constrained and Optimization Problems, Coconut Project, 2001, pp. 113-149.

Lhomme O., Gotlieb A., Rueher M., Taillibert P., "Boosting the interval narrowing algorithm", Proceedings of ICLP. MIT Press, 1996.

Lottaz C., Smith I.F.C., Robert-Nicoud Y., Faltings B.V., "Constraint-based support for negotiation in collaborative design", Artificial Intelligence in Engineering, Vol.14, 2000, pp. 261-280.

Merlet J.-P., "Projet COPRIN : Contraintes, OPtimisation, Résolution par INtervalles", Report, INRIA Sophia-Antipolis, Sept. 21st, 2001.

Moore R.E., "Methods and Applications of Interval Analysis". SIAM Studies in Applied Mathematics, 1979.
Sam J., "Constraint Consistency Techniques for Continuous Domains", Ph.D. Thesis number 1423, EPFL, 1995.
Scott M.J., Antonsson E.K., "Using Indifference Points in Engineering Decisions", DETC/DTM. Baltimore, Maryland, USA, September 10-13, 2000.
Thurston D.L., Liu T., "Design Evaluation of Multiple Attributes Under Uncertainties", International Journal of Systems Automation - Research and Application (SARA), Vol.1, 1991, pp. 143-159.
Ward A.C., Liker J.K., Sobek D.K., Cristiano J.J., "Set-based concurrent engineering and Toyota", DETC/DTM. Sacramento, California, 1994.
Wood K.L., Antonsson E.K., Beck J.L., "Computations with imprecise parameters in engineering design: background and theory", ASME Journal of Mechanisms, Transmissions and Automation in Design, No.March. 1989, pp. 616-624.
Yannou B., "Management of Uncertainty in Conceptual Design", Proceedings of International CIRP Design Seminar. Grenoble, France, May 12-14, 2003.
Yannou B., Simpson T.W., Barton R.R., "Towards a conceptual design explorer using metamodeling approaches and constraint programming", DETC/DAC. Chicago, Illinois, USA, September 2-6, 2003.

Ecole Centrale Paris
Laboratoire Génie Industriel - Grande Voie des Vignes 92295
Châtenay-Malabry, France
Telephone: (33) 1 41 13 16 05, Telefax: (33) 1 41 13 12 72
E-mail: yannou@lgi.ecp.fr