PROCESS OPTIMIZATION USING INTERVAL COMPUTATION

Daniel LEPADATU[1], Xavier BAGUENARD[2] , Abdessamad KOBI[1], Ridha HAMBLI[1] & Luc JAULIN[3]

[1] ISTIA – LASQUO (UPRES – EA 3858) , [2] ISTIA - LISA (FRE 2656 CNRS)
62, Avenue Notre Dame du Lac – 49000, Angers - France
Phone:(0033) – (0)241226580, Fax: (0033) – (0)241226521
E-Mail : daniel.lepadatu@istia.univ-anges.fr, kobi@istia.univ-angers.fr
[3] ENSIETA - Laboratoire E3I2 - 2, Rue François Verny – 29806, Brest, Cedex 9
E-Mail : xavier.baguenard@istia.univ-angers.fr, jaulinlu@ensieta.fr

Abstract: In manufacturing process, the quality of final products is significantly affected by both product design and process variables. However, historically tolerance research primarily focused on allocating tolerances based on product design characteristics of each component. This work proposes to expand the current tolerancing practices, and presents a new optimization method of tolerancing mechanical systems using interval computation for the prediction of system response. The proposed methodology is based on the development and integration of three concepts in process optimization: mechanical tolerancing, response surface methodology, and interval computation method. An industry case study is used to illustrate the proposed approach.

Résume: Dans le processus de fabrication la qualité de produit final est sensiblement affectée par des variables de conception de produits et de processus. Cependant, historiquement la recherche sur les tolérances s'est principalement concentrée sur l'utilisation des tolérances basées sur des caractéristiques de conception de produits pour chaque composant. Ce travail propose l'extension des pratiques de tolérancement courantes, et présente une nouvelle méthode d'optimisation pour le tolérancement des systèmes mécaniques en utilisant le calcul sur intervalle pour la prévision de la réponse du système. La méthodologie proposée est basée sur le développement et l'intégration de trois concepts par l'optimisation de processus: tolérancement mécanique, méthodologie de la surface de réponse, et méthode de calcul sur l'intervalle. Un cas industriel est utilisé pour illustrer l'approche proposée.

Keywords: mechanical tolerancing, response surface methodology, interval computation method, optimization, design of experiments.

Mots Clefs: tolérancement mécanique, méthodologie de la surface de réponse, calcul sur intervalle, optimisation, plans d'expériences.

1 – Introduction

Manufacturing operations are inherently imperfect in fabricating parts and assembly-products. Product imperfections were first described in the concept of part interchangeability and implemented in early mass production, which further led to the development of product tolerancing. Tolerancing is one primary means to guarantee part interchangeability. There is a significant body of literature related to tolerancing methods and its applications. Summaries of the state-of-the-art, the most recent developments, and the future trends in tolerancing research can be found in (Bjorke 1989 and Zhang 1997) as well as in a number of survey papers such as (Ngoi and Ong 1998, Voelcker 1998). Traditionally, tolerance analysis and synthesis in both stages have been studied in the context of product variables, i.e., they focused on part interchangeability. We feel that there is a tremendous need to further expand it to the interchangeability of manufacturing processes. This is becoming increasingly apparent with growing requirements related to manufacturer best practices, suppliers selection and benchmarking (where each supplier may use different process to manufacture the same product) or outsourcing. Tolerancing has the potential of being an important tool in such developments. We propose to extend the scope of tolerancing to explicitly include process variables in manufacturing processes.

It is therefore the purpose of this study to provide a design method for using Mechanical Tolerances (MT), Response Surface Methodology (RSM) and Interval Computation (IC) in process optimization. This method consist of combined the three concepts to obtain a powerful tool will be used especially to minimize the variability of the manufacturing process.

2 – Process optimization with design of experiments

The optimization of manufacturing process is essential for the achievement of high responsiveness of production, which provide a preliminary basis for survival in today's dynamic market conditions. Process optimization refers to manipulating the most important process variables to levels or settings that result in the best obtainable set of operating conditions for the system.

Response Surface Methodology (RSM) is a collection of mathematical and statistical technique for empirical model building (Cornell 1990). By careful design of experiments, the objective is to optimize a response (output variables) which is influenced by several independent variables (input variables).

Originally, RSM was developed to model experimental response (Box and Draper, 1987), and then migrated into modeling of numerical experiments. The difference is in the type of error generated by the response. An important aspect of RSM is the Design of Experiments (Box and Draper, 1987), usually abbreviated as DoE. The objective of DoE is the selection of the point where the response should be evaluated. In a traditional DoE, screening experiments are performed in the early stages of the process, when it is likely that many of the design variables initially considered have little or no effect on the response. The purpose is to identify the design variables that have large effect for further investigation.

To construct an approximate model that can capture interactions between N design variables, a full factorial approach (Montgomery 2001) may be necessary to investigate all possible combinations. A factorial experiment is an experimental strategy in which design variables are varied together, instead one at a time. If the number of design variables becomes large, a fractional of a full factorial design can be used at the cost of estimating only a few combinations between variables. This is called fractional factorial design and is usually used for screening important design variables. Genichi Taguchi, a Japanese engineer, proposed several approaches to experimental designs that are sometimes called "Taguchi Methods." "Taguchi" designs (Taguchi et al 1987) are similar to our familiar fractional factorial designs. However, Taguchi has introduced several noteworthy new ways of conceptualizing an experiment that are very valuable, especially in product development and industrial engineering, and we will look at two of his main ideas, namely Parameter Design and Tolerance Design (Ross 1988).

Once the variables having the greatest influence on the responses were identified, a special design was developed to optimize the levels of these variables. This design is a Box-Wilson Central Composite Design, (Box and Wilson 1951) commonly called 'Central Composite Design (CCD)', which contains an imbedded factorial or fractional factorial design having center points and being augmented by a group of 'star points' that allow estimation of curvature (Figure 1). If the distance from the center of the design space to a factorial point is ±1 unit for each factor then the distance from the center of the design space to a star point is $\pm\delta$ with $|\delta| > 1$. The CCD is the most popular class of designs used to fit a second-order model. In this case the model is defined as follows:

$$y = \beta_0 + \sum_{j=1}^{p} \beta_j x_j + \sum_{j=1}^{p} \beta_{jj} x_j^2 + \sum_{i=1}^{p} \sum_{i<j} \beta_{ij} x_i x_j \tag{2}$$

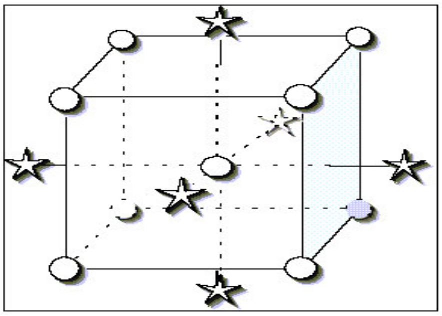

Figure 1. Central composite design for three factors.

Where: y is the response, x_i are the variables (factors) to be optimized, β_o is the independent term, and β_i, are the coefficients of the linear term β_{ij} are the coefficients of the quadratic term.

3 – Interval computation method

3.1 – Introduction

Interval computation was introduced to compute the solution in an guaranteed way. Actually scientific calculation made on computer doesn't work on real values but on truncated floats. The solution provided by interval analysis is to represent any real value by an interval containing it - see (Moore 1966) or (Alefeld et al 1983) for an introduction.
The arithmetic laws for interval calculation (Moore, 1979 and Neumaier, 1990) give a powerful tool to evaluate analytic function. Given two intervals $[x_1]$ $[x_2]$ arithmetic operation is defined by :

$$\begin{aligned}
&[x_1] + [x_2] = [a,b] + [c,d] = [a+c, b+d] \\
&[x_1] - [x_2] = [a,b] + (- [c,d]) = [a-d, b-c] \\
&[x_1]*[x_2] = [\min\{ac,ad,bc,bd\},\max\{ac,ad,bc,bd\}] \\
&1/[x_1] = [1/b , 1/a], 0 \notin [a, b]
\end{aligned} \tag{3}$$

These operations are an interval extension of real operations but not all the propriety can be transposed. For example x-x is considered as x-y with x and y independent variables. And if x is in $[-1,1]$ the expression will take $[-2,2]$ as result. This phenomena is called "dependency problem". The interval result of some expression grows with occurs of variables, however this is not true for all cases, e.g. $x+x$. Even if this problem was a serious issue, interval analysis may be used in many problems such as:
Global optimization (Hansen 1992), Determining roots of function (Kearfott 1997), Differential computation (Hammer et al. 1995), Robotics (Jaulin et al 2001), Bounded error estimation (Braems 2001). Other developments take into account discrete constraint propagation benefits (Cleary 1987 and Davis 1987). The interval constraints propagation provides new tools to suppress the dependency problem and new ways of considering problems.

A Constraint Satisfaction Problem CSP is defined by:

$$\begin{aligned}
&a \text{ set } V \text{ of } n \text{ variables } x_1,\ldots,x_n \text{ of } \Re \\
&a \text{ set } D \text{ of } n \text{ subset } [x_1],\ldots,[x_n] \text{ of } \Re\text{, called domains} \\
&a \text{ set } C \text{ of } m \text{ constraints relating variables } c_1,\ldots,c_m
\end{aligned} \tag{4}$$

On this CSP we can reduce domains with constraints propagation. The aim of constraints propagation is to give the smallest box for the domains including all the solutions close to the constraints. The solutions of this CSP are defined by the following set:

$$S=\{\mathbf{x} \text{ in } [\mathbf{x}], c_1(\mathbf{x}), c_2(\mathbf{x}),\ldots c_m(\mathbf{x})\} \tag{5}$$

An example of constraints propagation is given in the next section but many free solvers are available to characterize the solution set of a CSP - see (Baguenard et al, 2004), (Dao et al, 2004) and (Granvilliers 2002).

3.2 – Constraint propagation

Let a CSP be defined by the following constraint:

$$\begin{aligned}
&c_1: x_1+x_2 = x_3, \\
&x_1 \text{ in } [1,3],\ x_2 \text{ in } [0,2],\ x_3 \text{ in } [0,2].
\end{aligned} \tag{6}$$

The constraint is a relationship among variables. If variables are included in intervals, deductions can be made. For certain couples of points (x_1,x_2) we cannot find in the other interval $[x_3]$ a value to satisfy the constraint. These values are called "not consistent". There is no x_3 value for the couples $(x_1,x_2) = (3,2)$ and no (x_1,x_2) value for $x_3 = 0$, these values are not consistent values for this CSP.
The constraints propagation technique suppresses inconsistent values and reduces interval domains. In our CSP, domains obtained after constraint propagation are:

$$[x_1]*[x_2]*[x_3]=[1,2]*[0,1]*[1,2] \qquad (7)$$

The CSP implementation is defined by:

$$\begin{aligned}&[x_3]:=[x_3] \cap ([x_1]+[x_2]),\\ &[x_1]:=[x_1] \cap ([x_3]-[x_2]),\\ &[x_2]:=[x_2] \cap ([x_3]-[x_1]).\end{aligned} \qquad (8)$$

The constraint propagation operator for primitive constraints may also be defined as:

$$\begin{aligned}&c_2\text{: } x_1*x_2 = x_3,\\ &c_3\text{: } \sin(x_1) = x_2,\\ &c_4\text{: } \exp(x_1) = x_2...\end{aligned} \qquad (9)$$

All analytic expressions are a composition of + - * / operators or functions such as *sin, cos, exp,...* Therefore all constraints of the CSP are made of primitives constraints. Constraints propagation use this primitive's constraints to reduce variable's interval domains.

Constraints propagation is not the only method which contracts domains. An operator, called "contractor", may be defined for all techniques which reduce domains (Jaulin et al 2001).

3.3 – Estimation problem

To illustrate the estimation problem, we can consider the function (Equation 10) :

$$f(x)=a_1.exp(a_2.x). \qquad (10) \qquad\qquad S_1=\{\mathbf{a},\mathbf{x_i} \text{ in } [\mathbf{x_i}],f(\mathbf{x_i}, \mathbf{a}) \text{ in } [\mathbf{y_i}]\}. \qquad (11)$$

In order to estimate a_1 and a_2 values which are in the set we consider (Equation 11). In figure 2 the interval's domains for x_i and y_i are represented by gray boxes. A solution included in the set S is given by the dark curve. This curve corresponds to a couple of points (a_1,a_2) which represents a solution for our estimation problem. The dotted line is a non solution. In this case $[f(\mathbf{x_2}, \mathbf{a})]$ is not included in $[y_2]$. In our problem, we need to obtain an interval value for each $\boldsymbol{a_i}$, which will enable us to chose our value.

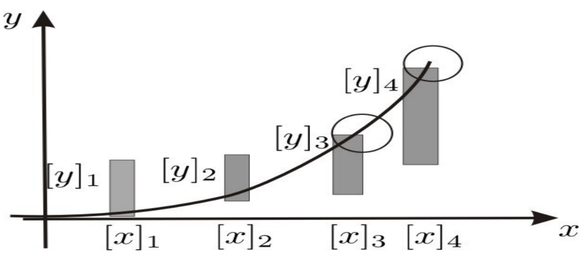

Figure 2. Parameter estimation for an exponential function.

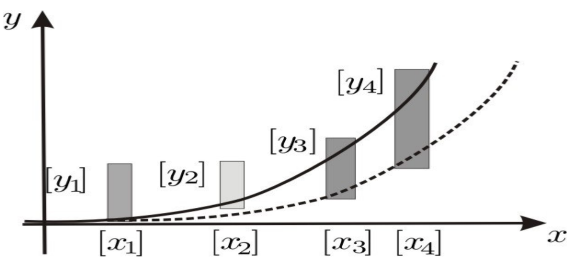

Figure 3. Example of estimation with a non robust solution.

You can also try to find a more specific set where constraints are satisfied for all the values of $[x_i]$ (Ratchan 2000). The set of solution is:

$$S_2=\{\mathbf{a},\ \forall\ \mathbf{x} \text{ in } [\mathbf{x}],f(\mathbf{x}) \text{ in } [\mathbf{y}]\}. \qquad (12)$$

A non robust value is drawn in figure 3. In the circle (Figure 2) there is a certain number of values which are not included in $[y_i]$. This set of values S_2 are included in S_1. In our application we are more interested in this particular type of set because, having the interval solution, we can choose a value for x_i.

3.3 – Proposed algorithm

At first a local method is applied, providing a local solution for a_i. It gives us an information about the initial domains of $[a_i]$. With this information and interval domains for x_i and y_i, we may write the CSP file. The next step reduces intervals' domains in a forward-backward propagation on all constraints. This step is called *CS([a],[x],[y])*

Secondly we bisect one of a_i intervals domain. We made an interval assessment of *f(a,x)* for each part of the domain and selected one of them. The used criterion (Figure 4.) is a function which compares two boxes, e.g. $[f([a]_1]$ and $[y]_i$. The result is the largest distance separating the two boxes. If we apply the criterion on $[a]_i$ evaluation, we obtain two easily comparable distance values.

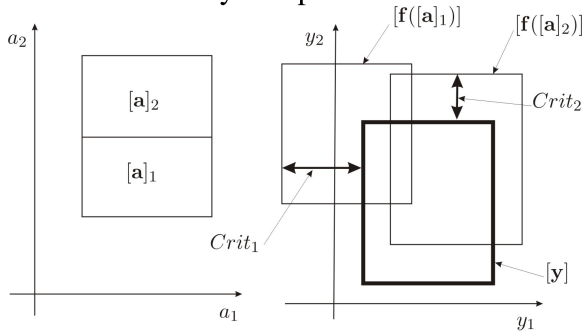

Figure 4. Criterion to select a_i box.

If the evaluation $[f([a]_1)]$ is in $[y_i]$ for all *i,* the criterion is negative. We have found a box which is a solution for $[a_i]$.
Of course the algorithm chooses one box $[a]_i$ and may have lost solutions, but the criterion depends on initialization. Some developments can be made to reduce this solution's loss.
This first algorithm was made to show interval methods possibility for this problem and further implementations can be made. For example some variables occur more than once. The dependency problem can cause overestimation in interval evaluation, and the contraction method doesn't give the smallest box. We can use box contraction to enforce contraction (Benhamou et al 1994). Progression in contraction can also be done with specific contractors such as the gauss elimination method.

4 – Mechanical tolerancing

The inherent imperfections of manufacturing process cause a degradation of product characteristics, and therefore of product quality (Dantan et al. 2003). "Tolerance" is a method used to describe variability in a product or production process. It defines the acceptable ranges in the actual performance of a system or its components, across one or more parameters of interest, under the conditions considered during design, for which the system or components are fit for purpose, i.e., meet the specifications and/or customer expectations. Tolerances historically provide the means for communication between product and process designers (Milberg 2002). Higher precision would mean lower tolerance and better machines are needed to manufacture the parts and thus, this will increase the cost to manufacture the parts. Tolerance is a key factor in determining the cost of a part. As mentioned earlier lower tolerance will results in a higher cost of producing the parts. The relationship between tolerance and manufacturing cost is shown in the figure 5.

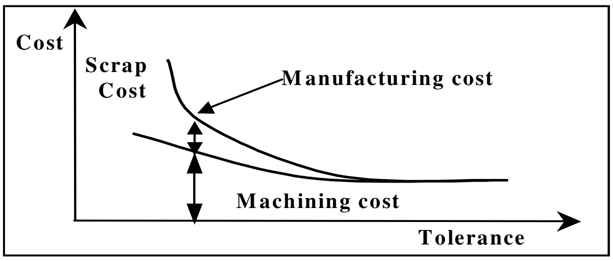

Figure 5. Manufacturing cost versus mechanical tolerance.

The manufacturing cost is divided into machining and scraps cost (Figure 5).
- The machining cost is the cost of first producing the part.
- The scrap cost is the cost encountered due to rejecting some parts that fall outside the specified tolerance range.

Generally, product or process are considered in conformity when it is in one acceptation interval (tolerance) (Sergent et al. 2003). Tolerance analysis views component-related tolerances as a range of values in terms of variation from a nominal value. Tolerance analysis takes a given set of component

tolerances, usually based on designer experience or standards, and calculates the resultant variation in the assembly. Through iteration, component tolerances are tightened to meet assembly tolerances, establishing both the product and process design requirements. In contrast, tolerance allocation looks at a range of component designs around a functional or assembly description to absorb the variability. Tolerance allocation is used to maximize quality, minimize production cost, or both. The result can be looser component tolerances and better matching of product and process (Trabelsi et al. 2000, Gerth 1997). Tolerance allocation creates robustness.
In order to minimize the scraps cost, we propose a new method which increases the acceptation interval of the assembly parts in manufacturing process of mechanical pieces.

5 – Proposed approach

This work aims at defining a new method of optimization that will use three concepts:

- Response Surface Methodology
- Interval Computation Method
- Mechanical Tolerancing

Actually, we will tolerate every level of parameters $X_{i_{\min}^{\max}}$ with specific bilateral tolerances $\pm\Delta_i$, which will later allow the usage of the proposed Interval Computation algorithm in order to obtain what one may call "Interval Response Surface" (IRS).
The obtained equation of the IRS will allow us to choose several sets of "parameter games" so as to make the system more flexible. It is very important to mention the fact that for all the sets of "parameter games" the response to be optimized will always remain "admissible". That means that in an acceptance interval of the response established by experts or by engineers a priori while respecting specifications, the response will no longer represent a single value "target", but an interval.
Specifications often take the shape of a target value (the nominal value) m with the bilateral tolerance Δ.
It is an error to think that such a specification means that all values included between $m - \Delta$ and $m + \Delta$ will also have the same low quality.
Therefore the “engineering of the target” doesn't eliminate the need of tolerances. The existence of tolerances will also confer a certain flexibility to the manufacturing process and therefore will increase the chances of products’ acceptance within the bearable limits so as to be functional.
This new method will bring flexibility in adjusting parameters to find the optimum of a manufacturing process specifically for multireponse optimization where the probability to “play” on the sets of parameters to find an acceptable optimum is not as high.

6 – Application

In order to illustrate the proposed approach we present the example (Lepadatu et al. 2004) of an extrusion process optimization problem which is currently applied in mechanical manufacturing industry.
Recently, the extrusion process (Figure 6) has acquired a fundamental role in metal forming and many researches proposed different design techniques (Gierzynska-Dolna et al 2003, Arif A.F.M. et al 2003) for increasing the performance and the service life tool.

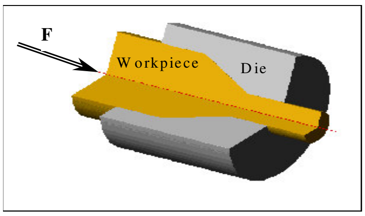

Figure 6. Metal extrusion process.

Generally, the die lifetime is determined by the stress state of die in working conditions and by the processed die material proprieties. The estimation of tool life (fatigue life) in extrusion operation is important for scheduling tool changing times, for adaptive process control and tool cost evaluation. The cost of forming tools usually covers a substantial amount of the forming parts’ total manufacturing cost

(Yi-Che Lee et al. 2000). More details concerning this application may be found in (Lepadatu et al. 2004). This example used a Central Composite Design (CCD) with 3 variables, that is, X_1 (Angle of the die) , X_2 (Friction), X_3 (Temperature). (Table 2). This design provides five levels for each design variable ($\pm\delta$, ±1, and 0 - Table 2). Each level for each corresponding parameter is written in interval form in term of real values (Table 2). It is important to mention that the two interval limits are bilateral mechanical tolerances ($\pm \Delta i$) for each parameter. The two responses for this work are Y_1 = Maximum deformation and Y_2 = Lifetime of die (Table1).

Table 1. Interval Design Matrix of CCD.

Runs	X_1	X_2	X_3	Y_1	Y_2
1	[0,063;0,077]	[20,7; 25,3]	[523; 640]	[1,13 ;1,39]	[4781 ; 8069]
2	[0,063;0,077]	[20,7; 25,3]	[737; 901]	[1,21; 1,48]	[33370,47181]
3	[0,063;0,077]	[28,8; 35,2]	[523; 640]	[1,38 ;1,69]	[7485,11784]
4	[0,063;0,077]	[28,8; 35,2]	[737 ; 901]	[1,45 ;1,78]	[34936,50745]
5	[0,117;0,143]	[20,7; 25,3]	[523; 640]	[1,62 ;1,98]	[8874,14303]
6	[0,117;0,143]	[20,7; 25,3]	[737; 901]	[1,64 ;2,01]	[30834,40048]
7	[0,117;0,143]	[28,8; 35,2]	[523; 640]	[1,75 ;2,13]	[8870,15486]
8	[0,117;0,143]	[28,8; 35,2]	[737; 901]	[1,74 ;2,12]	[30870,40098]
9	[0,045;0,055]	[20,7; 25,3]	[630; 770]	[1,15; 1,40]	[13899,22859]
10	[0,135;0,165]	[20,7; 25,3]	[630; 770]	[1,73 ;2,11]	[12513,18786]
11	[0,09 ; 0,11]	[24,8; 30,2]	[630; 770]	[1,52 ;1,86]	[15828,22150]
12	[0,09 ; 0,11]	[24,8; 30,2]	[630; 770]	[1,70 ;2,08]	[17089,23513]
13	[0,09 ; 0,11]	[18 ; 22]	[450; 550]	[1,50 ;1,83]	[8254,16464]
14	[0,09 ; 0,11]	[31,5;38,5]	[810; 990]	[1,52 ;1,85]	[60295,93465]
15	[0,09 ; 0,11]	[24,8; 30,2]	[630; 770]	[1,49 ;1,82]	[17885,24840]

Table 2. Coding of the parameters – interval form.

Parameter	Levels				
	$-\delta \pm \Delta_i$	$-1 \pm \Delta_i$	$0 \pm \Delta_i$	$1 \pm \Delta_i$	$\delta \pm \Delta_i$
X_1	[0,045; 0,055]	[0,063 ; 0,077]	[0,09 ; 0,11]	[0,117 ; 0,143]	[0,135 ;0,165]
X_2	[18 ; 22]	[20,7; 25,3]	[24,8; 30,2]	[28,8; 35,2]	[31,5;38,5]
X_3	[450 ; 550]	[523; 640]	[630; 770]	[737 ; 901]	[810 ; 990]

Table 3. Equations for ORSM and IRSM

The Model	Y_1 *ORSM*	Y_2 *ORSM*	Y_1 ***IRSM***	Y_2 ***IRSM***
C	1,657560	20890,26	**[1.45; 1.72]**	**[20345; 21992]**
X1	0,206741	-366,02	**[0.12; 0.29]**	**[-498; -213]**
X1X1	-0,027817	-2264,27	**[-0.07; -0.011]**	**[-2487; -2046]**
X2	0,081178	548,51	**[0.05; 0.11]**	**[405; 689]**
X2X2	0,040066	-986,39	**[0.01; 0.12]**	**[-1123; -902]**
X3	0,015740	14912,80	**[0.005; 0.11]**	**[14030; 15067]**
X3X3	-0,000416	6405,12	**[-0.001,-0.0001]**	**[6304; 6699]**
X1X2	-0,036875	-588,68	**[-0.012 ;-0.01]**	**[-712; -497]**
X1X3	-0,019875	-1955,57	**[-0.09; -0.001]**	**[-2995; -1067]**
X2X3	-0,005125	-152,52	**[0.009 ;-0.001]**	**[-252; -51]**

Ordinary Least Squared (OLS) estimation technique was first applied to the initial data (Lepadatu et al. 2004) to develop the Ordinary Response Surface Models (ORSM) for each response Y_i. The equations for generated models (in terms of coded factors) are represented in table 3.

Using the proposed algorithm (see section 3.4) for the data in table1, the Interval Response Surface Models (IRSM) are developed for each response Y_i. The equations for generated models (in terms of coded factors) are as follows (Table 3).

For example the equation 12 (in interval form) of the Interval Response Surface for the die lifetime is:

$$\boldsymbol{Y}_2 = [20345; 21992] + [-498; -213]\ X_1 + [-2487; -2046]\ X_1\ X_1 + [405; 689]\ X_2 + [-1123; -902]\ X_2\ X_2 + [14030; 15067]\ X_3 + [6304; 6699]\ X_3 X_3 + [-712; -497]\ X_1 X_2 + [-2995; -1067]\ X_1 X_3 + [-252; -51]\ X_2 X_3 \qquad (12)$$

These equations (Table 3) allow the obtaining of a Tolerancing Response Surface, called *Interval Response Surface*, and represents a new manner for making many products accepted in the manufacturing process optimization.

7 – Conclusions

This paper has described a manufacturing process optimization method that combined the Response Surface Methodology, Interval Computation Method and Mechanical Tolerancing. In this work we proposed a new method to obtain a new Response Surface Methodology called Interval Response Surface used in the process optimization. Using this method more final pieces are produced and accepted in the manufacturing process optimization.

References :

Alefeld G. and J. Herzberger (1983), *Introduction to Interval Computations.* Academic Press.

Arif A.F.M. et al (2003), A study of die failure mechanism in aluminum extrusion, JMPT 134.

Baguenard X. , M. Dao and L. Jaulin (2004) Solver Interval Peeler, available at, http://www.istia.univ-angers/~baguenar

Benhamou F. , D McAllester and P.Van Hentenryck (1994) CLP Intervals Revisited, Proceedings of the International Logic Programming Symposium, (124-138) Ithaca,NY.

Bjorke, O., (1989), Computer Aided Tolerancing, Tapir Publishers, Trondheim, Norway.

Box, G. E. P., and Draper, N. R., (1987), Empirical Model Building and Response Surfaces, John Wiley & Sons, New York, NY.

Box G.E.P. and Wilson K.B. (1951), On the experimental attainment of optimum conditions, J. Royal Stat. Soc. B 13, p: 1–45.

Braems and L. Jaulin and M. Kieffer and E. Walter (2001) Set Computation, computation of Volumes and Data Safety, to appear in the Proceedings of SCAN

Cleary J. G. (1987) Logical arithmetic. Future Computing Systems, 2, 2, (125-149).

Cornell J.A. (1990), *How to Apply Response Surface Methodology*'', vol. 8, ASQC, Wisconsin.

Dantan Jean-Yves et al. (2003), Spécification géométrique fonctionnelle des engrenages par frontière virtuelle, CPI'2003 – Mekhnès Maroc

Dao M. , X. Baguenard and L. Jaulin (2001) Solver Proj2D, available at, http://www.istia.univ-angers/~dao/Proj2DV2.zip/

Davis E., (1987) Constraint propagation with interval labels. Artificial Intelligence. 32, 3, (281-331)

Gerth, R. (1997). Tolerance Analysis: A Tutorial of Current Practice. Chpt. 4 in *Advanced Tolerancing Techniques*. Zhang, G. (ed.). NY: Wiley.

Gierzynska-Dolna M et all (2003), Some aspect of modeling of metal forming process, Computer and Structure Journal 81.

Granvilliers L. (2002).Realpaver available at, http://www.sciences.univnantes.fr/info/perso/permanents/granvil/realpaver/

Hammer R. , M. Hocks, U. Kulisch, and D. Ratz. (1995), *C++ Toolbox for Verified Computing*. Springer-Verlag, Berlin.

Hansen E., (1992), *Global Optimization Using Interval Analysis*, Marcel Dekker, Inc., New York

Jaulin L., et al (2001). *Applied Interval Analysis*, with Examples in Parameter and State Estimation, Robust Control and Robotics. Springer-Verlag. London

Kearfott R. B. (1997), *Rigorous Global Search: Continuous Problems.* Kluwer Academic Publishers, Dordchet.

Lepadatu D., Kobi A., Hambli R. and Barreau A. (2004), Lifetime Multiple Response Optimization of Metal Extrusion Die. The 51st Reliability and Maintainability Symposium, 24-27 January 2005 Alexandria - Virginia

Milberg Colin , Iris D. Tommelein, and Thais Alves, (2002), Improving design fitness through tolerance analysis and tolerance allocation, 3rd International Conference on Concurrent Engineering in Construction 1-2 July 2002 – University of California, Berkeley.

Moore R. E. (1966), *Interval Analysis*. Prentice Hall, Englewood Cliffs, NJ.

Moore R., (1979), Methods and Applications of Interval Analysis, SIAM Studies in Applied Mathematics, Philadelphia

Montgomery Douglas C., 2001, *Design and analysis of experiments*. – 5th edition by Wiley & Sons, Inc., New York.

Neumaier A., (1990), Interval Methods for Systems of Equations, Cambridge University Press

Ngoi, B.K.A and Ong, C. T., (1998), "Product and Process Dimensioning and Tolerancing Techniques: A State-of-the-Art Review," International Journal of Advanced Manufacturing Technology, 14, 910-917.

Philip J. Ross 1988, *Taguchi Techniques for Quality Engineering*, McGraw-Hill, Inc. USA

Ratschan S., (2000) Approximate Quantified Constraint Solving ({AQCS}) 861-880, Available_at:_http://www.risc.unilinz.ac.at/research/software/AQCS

Sergent Alain, Daniel Duret, Maurice Pillet (2003) La Tolérance Type, CPI'2003 – Mekhnès Maroc

Taguchi, G. and Konishi, S., (1987), *Orthogonal Arrays and Linear Graphs*, Dearborn, MI, ASI press.

Trabelsi, A. and Delchambre, A. (2000). Assessment on Tolerance Representation and Tolerance Analysis in Assemblies. Concurrent Engineering: Research and Applications v. 8 (4). Technomic Publishing Co., Lancaster, PA, Dec., pp. 244-262.

Voelcker, H.B., (1998), "The Current State of Affairs in Dimensional Tolerancing," Integrated Manufacturing Systems, 9, 205 – 217.

Yi-Che Lee et al (2000), Failure analysis of a cold-extrusion punch to enhance its quality and

Zhang, G., (1996), Simultaneous Tolerancing for Design and Manufacturing, International Journal Of Production Research, 34, 3361-3382.