Process Tolerancing: A new statistical tolerancing method for industrial processes not daily adjustable in mass production.
Proposal of an improvement to Wayne Taylor’s method.

Dr. Jean-Marc JUDIC

Faurecia SA; Route d’Etampes; BP89; 91152 ETAMPES Cedex
01 69 92 34 21 / 06 62 94 26 81
E-mail1: jean-marc.judic@faurecia.com
E-mail2: jeanmarc.judic@sfr.fr

Abstract: Numerous authors have demonstrated limits and risks of usual tolerancing methods. Imposing expensive tolerances and 100% checking without risk assessment, Worst Case is definitively not appropriate to mass production. Common root sum squares statistical tolerancing enlarges significantly the tolerances, but becomes severely dangerous when hidden assumptions are not perfectly fulfilled. The problem is mainly due to the centering of components, which is never fully guaranteed by non-adjustable processes such as blanking, stamping or moulding, or when average is hopelessly variable during production time. “Inertial tolerancing” from Maurice Pillet improves statistical tolerancing robustness, but is mainly dedicated to industries using adjustable processes, as cutting. Another innovative method, named Process Tolerancing, from Wayne Taylor, based on the same analysis, is much better adapted to industrial processes requiring operational margins for mean centering. We propose here an improvement to this method and present associated adaptations to SPC tools and routines.

Key words: Statistical tolerancing, Product and process engineering, SPC, Quality.

1 Acknowledgement

I would like first to express my gratitude to M. Pillet, B. Anselmetti and W. Taylor for their high contributions in the field of tolerancing. Without their works, this paper, providing only a brick to the edifice, would never emerged.

2 About the industrial problem of tolerances.

Tolerancing is a key task for mass production industry, linking process and product engineering sides into a single goal: customer satisfaction. From design office to manufacturing floor, two contradictory points of view are in conflict. Product engineers really don’t need tolerances and could expect only parts at the nominal. Unfortunately, for technical and economical reasons, to produce thousands of parts perfectly on the target is impossible. Allocation of tolerances is necessary to allow manufacturers to produce “non perfect” but “acceptable” parts.

The difficulty in this first step of tolerancing activities is clear: how to enlarge tolerances on components to satisfy manufacturers without taking the risk to displease customers with a wrong quality level. To be well done, this task requires the building of high level technical models to predict performances and to identify most robust design choices for the product as for the process. Without scientific effort, this research of an optimal compromise is in risk to stay a dark negotiation and not the relevant collaboration between product and manufacturing engineers.

The second tolerancing is the tolerances specification and representation on drawings. According to its consequences, this tricky task is probably not sufficiently valued by engineers. Last task is tolerances validation, and difficulties appear for plant’s operators under cost and time constraints. A first issue is to measure the right thing on one part, a second one is to define when and how many parts to measure. Finally, operators must decide to accept or to reject, sometime a part, a batch, a truck, sometime a tool… Often, if not generally, the decision must be taken rapidly without any upper view on the final product.

As describe below, currents practices are not always satisfactory while theoretical advances are well established, and improvement perspectives really exist.

3 Tolerancing methods background

3.1 Problem description

The starting point in Tolerancing is to identify the criteria on which ones parts and process variability may have an effect on customer satisfaction. Conventionally we name Y these product performance criteria, characteristics generally measurable by the customer on final product. On a Y, customer specifications apply, with tolerances bounds and associated quality requirements (Cpk or ppm level defined according the severity of unconformity effects). A non conformity on a Y is a defect for the customer, with potential associated penalties, returns or recalls if detected. According to product functioning and process working, a Y results from several, sometimes many, parameters, as
components characteristics, tooling characteristics, operators, external conditions... Without distinction of type, we usually name these parameters, the X’s.

The first task for product and manufacturing engineers is to identify or approximate the transfer function:

\[ Y = F(X_i) \quad i = 1..n \]  

(1)

Even if F function is not linear, when X’s variations are small, a linear approximation provides an efficient prediction of performance variations around the target. The only requisite is to well recalculate the coefficients each time a nominal value changes. Agreeing also that a design with significant nonlinearities or interactions will never be the optimal solution to ensure the robustness of a product, the case of linear behaviours covers a wide and attractive domain for industry. In the following development, we consider to be in the linear approximation validity domain, and we have:

\[ Y = a_0 + \sum_{i=1}^{n} a_i X_i \]  

(2)

with \( a_i \) coefficients generally named sensitivities.

3.2 Process Capability ratio

For a performance criteria Y, we designate \([Lsl, Usl]\), the customer specifications limits and \( T_Y = (Lsl+Usl)/2 \) the target. If \( \mu_Y \) is the observed mean and \( \sigma_Y \) the standard deviation of the production, we calculate the following capability ratios:

\[ Cp = \frac{Usl - Lsl}{6\sigma_Y} \quad Cpl = \frac{\mu_Y - Lsl}{3\sigma_Y} \quad Cpu = \frac{Usl - \mu_Y}{3\sigma_Y} \]

and then

\[ Cpk = \min(Cpl, Cpu) = Cpk = \frac{|T_Y - \mu_Y|}{3\sigma_Y} \]  

(3)

Cp represents process potential ability to produce parts inside the tolerance width. Cpk, the most popular process capability indices, shows process performance integrating the centering.

When Y distribution is normal, the defect ratio in ppm can be directly calculated according to Cpl and Cpu values:

\[ ppm = \left[ \text{normcdf}(-3.Cpl) + \text{normcdf}(-3.Cpu) \right] 10^6 \]  

(4)

with \text{normcdf}(f) the normal standard cumulative function.

By knowing only Cpk value, we only can provide an interval:

\[ \text{normcdf}(-3.Cpk).10^6 < \text{ppm} \leq 2.\text{normcdf}(-3.Cpk).10^6 \]

i.e. with Cpk=1, we get 1350 < ppm ≤ 2700.

3.3 Worst-Case tolerancing (WCT)

We consider specifications done for each Xi by:

\[ X_i = t_i \pm \Delta_i \]  

(5)

with \( \Delta_i \) the tolerance around the target \( t_i \) for \( X_i \)

Using WCT, the resulting tolerance on Y is ([P2] & [E1]):

\[ Y = t_Y \pm \Delta_Y \]  

(6)

with \( t_Y = a_0 + \sum_{i=1}^{n} a_i t_i \) the resulting target

and

\[ \Delta_Y = \sum_{i=1}^{n} |a_i| \Delta_i \]  

the resulting half tolerance

The designer works in inverse way by specifying targets and tolerances on the X’s to get the desired result on the Y. With WCT, the computation is simple, and the decision making doesn’t require statistics: with good parts, we are sure to produce good assemblies. Unfortunately, the cost of this advantage is heavy.

First, average tolerance on the X’s becomes very small when the number of X increases. For a classical linear stack-up with sensitivities at 1 or -1 we get:

\[ \overline{tol_X} = tol_Y / n \]  

(7)

The second weakness of WCT is to be unable to provide a statistical assessment on resulting performance on the Y and offers any risk management opportunities. Operator in the plant has no alternative than to scrap bad parts, even if he knows by is daily experience that a shaft a little too big easily enters in median hole. WCT forgets that bad parts may produce good assemblies and considers the worst combination as the most probable event.

To transfer the conformity decision on a Y into a binary decision on its X, without upper view on the complete system and without probabilistic calculations, can’t be the right method for our industry in the 21st century. The following example of the tube in a U-shape illustrates the unnecessary waste potentially caused by obsolete rules and routines.
This figure 2 shows that a large domain out off Worst-Case conformity area is able to provide good assemblies...

Imagine we receive the following batches of gussets & tubes:

<table>
<thead>
<tr>
<th></th>
<th>mean</th>
<th>std</th>
<th>Cpl</th>
<th>Cpu</th>
<th>scrap</th>
</tr>
</thead>
<tbody>
<tr>
<td>L</td>
<td>20.5</td>
<td>0.0866</td>
<td>1.15</td>
<td>0.38</td>
<td>12.4%</td>
</tr>
<tr>
<td>D</td>
<td>20.2</td>
<td>0.0500</td>
<td>2.67</td>
<td>0.00</td>
<td>50.0%</td>
</tr>
<tr>
<td>Y=L-D</td>
<td>0.3</td>
<td>0.100</td>
<td>1.00</td>
<td>1.67</td>
<td></td>
</tr>
</tbody>
</table>

According to the poor quality on components, WCT is going to scrap 12.4% of the gussets, and 50% of the tubes, without any benefits for the customer! And the solution is of course not to stop the controls and to produce whatever the parts are. Bad parts may produce good assembly, but the only way to manage scientifically the situation is to introduce risk management by using in the right way statistics.

On the other hand, even if Cpk targets on components are achieved, WCT doesn’t offer robustness on the final performance whenever Go/No-Go gauges are not fully operational to filter the production at 100%.

The following graph shows what may happen using WCT when manufacturers prefer to control samples to assess their quality in place to control all parts and to scrap all non conform ones. For this simulation, we made the hypothesis to have a centered production for the tubes with a Cp=Cpk=1, and to have also Cpk=1 for the gussets with a mean bias from 0 to 0.18 (Cp=1 to 10). Then we asses the resulting performance on Y=L-D.

![Figure 3](image-url)

**Figure 3 : Worst-case is not insensitive to off entering**

Without to introduce here the discussion about sampling risk management, we can be sure that only few manufacturers will be worried to get a Cp=6 with a Cpk at 1. Here, for this, we may get 278ppm on safety criteria side (Y<0.8). Some quality engineers could try to correct this trouble by cascading on components the required Cpk target on the Y over a Worst-Case tolerance allocation. This solution is of course an economical disaster, reducing manufacturer margin and cancelling the imaginary advantage of worst-case tolerancing: to not require statistics!

3.4 Statistical tolerancing

Many authors have very well explained the “economical advantage” of standard statistical tolerancing and illustrated its risks and limits [A1], [B1], [C1], [E1], [G1], [P1], [P2]. After them, it is very difficult to be original. Nevertheless, according to the difficulty to convince and to make the practices to change we propose our bricks to the demonstration.

The main reason of statistical tolerancing success is clear. For a 1D stack up of n components, this method allows to enlarge the average tolerance on components by vn. A stack of 9 parts, we multiply worst-case tolerances on average by 3! How to resist?

So often, engineers only remember the most popular formula of statistical tolerancing, the famous Root Sum Squares formula giving the standard deviation on the Y:

\[
\sigma_Y = \sqrt{\sum a_i^2 \sigma_i^2} \quad (\text{with } \sigma_i : \text{X_i standard deviation}) \quad (10)
\]

And they think that the mistake comes from this formula. However, the only assumption required here is the independence of the X_i, and in anyway a discussion about the normality of distributions, even if the achievable prediction knowing \(\sigma_Y\) depends on the resulting distribution on Y.

A first most important trouble appears when we want to transfer this formula talking about standard deviations into terms of tolerances, and many pitfalls exist the X_i and the Y are not specified with identical capability coefficients [G1].

But, as well explain by previously cited authors, the main issue comes from another and surprising side: the calculation of the mean on the resulting Y:

\[
\mu_Y = \mu_o + \sum_{i=1}^{n} a_i \mu_i \quad (\text{with } \mu_i : \text{mean of X_i}) \quad (11)
\]

This second formula theoretically requires any hypothesis for a linear criterion, but the problem comes when we introduce assumptions about components means \(\mu_i\) in place to use observed values.

The assumption done in regular statistical tolerancing tools is that components’ productions are perfectly centered. And the Achilles’ hell on the method is hidden in a so simple formula \(\mu_Y = (Lsl_D + Usl_Y) / 2\).

This inference is wrong because it neglects production biases, an unfortunately biases exist and may produce disastrous effects. Using usual statistical tolerancing method, it is possible, when centering hypothesis is not perfectly fulfilled, a very high of defect on assembly with fully acceptable productions for components.

To demonstrate this S. Bisgaard & S. Grave [B1] have build the example of the stack of 10 washers and M. Pillet [P2] has created a very nice simulator for a stack of 3 components. We propose here an adaptation of Bisgaard’s example with 4 washers only to show with easier calculations that the mistake doesn’t need 10 components to appear.

\[
X = 1 \pm 0.12 \quad \text{with CpkX=1}
\]

The usual statistical calculation gives a standard deviation for y equal to:

\[
\sigma_Y = \sqrt{4 \times \left( \frac{0.12}{3.Cp_X} \right)^2} = 0.08
\]

We have \(\sigma_X \leq 0.12 \leq 0.04\), and we get \(\sigma_Y \leq 0.08\).
confirming the tolerance interval on Y of ±0.24 for a Cpk=1.

Imagine now we receive a truck of washers with a mean at 1.06, and a standard deviation of 0.02 (without sampling uncertainty consideration for the moment).

Usual quality ratios are:
- $C_p = 0.12/(3\times0.02) = 2$
- $C_{pk} = (1.12 - 1.06)/(3\times0.02) = 1$

And the Quality Manager is naturally happy!

Let’s see what statistics are in reality able to predict using the formulas (10) & (11): $\mu_Y = 1.06 + 1.06 + 1.06 = 4.24$ $\sigma_Y = (0.02^2 + 0.02^2 + 0.02^2)^{0.5} = 0.12728$ Then we get $C_{pk} = (4.24 - \mu_Y)/(3\sigma_Y) = 0$!

We are going to get a scrap of 50% on assemblies with fully acceptable component’s productions!

The next example, reusing the tube & gusset welding in figure 1, is going to illustrate more graphically this lake of robustness of usual statistical tolerancing.

We consider here that the specifications on the Y are the following:
- $Y_{min} = 0$ with $C_{pl} \geq 1.333$ (process issue)
- $Y_{max} = 0.8$ with $C_{pu} \geq 1.677$ (safety issue)

Using usual statistical tolerancing, it is possible to propose the following specifications:
- for the tube $D = 20.05 \pm 0.3$ with $C_{pk} = 1.677$
- for the gusset $L = 20.35 \pm 0.3$ with $C_{pk} = 1.677$

For $Y = L-D$ we verify that the targeted mean is 0.35 and the standard deviation $\sigma_Y \leq \sqrt{0.06^2 + 0.06^2} = 0.08485$ So, we get: $C_{pl} \geq (0.35 - 0)/(3\times0.08485) = 1.375$ $C_{pu} \geq (0.8 - 0.35)/(3\times0.08485) = 1.77$

This design seems to be really acceptable!

As well illustrated by Pillet [P1] with his “off-centering” simulator, we are going to see now the consequences of means biases on our 2 components.

Imagine that we receive the following productions:
- for the tube $\mu_D = 19.8$ & $\sigma_D = 0.02$
- and for the gusset $\mu_L = 20.55$ & $\sigma_L = 0.02$

It is easy to verify that both productions satisfy the Cpk target of 1.667 (rem : with a Cp=5 !). And for the resulting criteria we get: $\mu_Y = 20.55 - 19.8 = 0.75$ and $\sigma_Y = 0.02 \times \sqrt{2} = 0.0282843$ and then $C_{pu} = (0.8 - 0.75)/0.0282843 = 0.59$ !

For a targeted Cpk at 1.667 in input, we get defect ratio level at 38550 ppm for a safety characteristic!

This result demonstrates that usual statistical tolerancing method is really not robust.

On the lower bound, we can get more demonstrative results without to need a Cp=5 on components.

With the following productions:
- for the tube $\mu_D = 20.2$ & $\sigma_D = 0.03$
- and for the gusset $\mu_L = 20.15$ & $\sigma_L = 0.03$

We fulfill the Cpk target (1.667) with a Cp of 3.33. And we get on Y : $\mu_Y = 0.05$ & $\sigma_Y = 0.03 \times \sqrt{2} = 0.042426$ And then $C_{pl} = (0.8 - 0.05)/0.042426 = 0.39$ !

Corresponding to a defect ratio nearly 12% !

The next figures illustrate this fundamental issue. The first represents initial tolerance allocation centred productions assumption and the second shows what happen when undesired off-centerings appear.

Statistical tolerancing enlarges worst case tolerances; by the way it becomes possible to get bad assemblies with good parts, then initial calculation done for tolerance allocation fall down when off-centering appear.

3.5 Inertial tolerancing from M. Pillet

In agreement with Taguchi loss function and to the goal to reduce the cost of production variability, the idea of Maurice Pillet is to apply the tolerancing on the loss factor in place to the parts variation range [P2].

Taguchi loss function is: $L = k \times \Sigma (x - \text{target})^2$, where $k$ is an economical factor. For a batch of $N$, it is demonstrated that we get

$$L = k \times \left( \sigma^2 + (t - \mu)^2 \right) = k \times \left( \sigma^2 + \delta^2 \right)$$

with $t$, the target, $\mu$ the mean of the $x_i$, $\sigma$ their standard deviation, and $\delta = t - \mu$.

From this point of view, to reduce non quality costs, it is clearly the right job to want to contain this factor $(\sigma^2 + \delta^2)$

For this reason, M. Pillet suggests to define an inertia as $I = \sqrt{\sigma^2 + \delta^2}$ and to specify a criteria by its target, and its...
The best illustration of this tolerancing method is the following figure [P2]:

![Diagram](image)

**Figure 6 : Inertial tolerancing acceptance zone**

If all components $X_i$ are just on the border of the semi-circle, so have all their maximal inertia we get for $X_i$ a standard deviation $\sigma_i = \sqrt{\sum_{i=1}^{N} a_i^2 I_i^2 + \delta_i^2}$ where $\delta_i$ is the bias on $X_i$ mean.

We get $\delta = \sum_{i=1}^{N} a_i \delta_i$ and $\sigma = \sqrt{\sum_{i=1}^{N} a_i^2 I_i^2 - \sum_{i=1}^{N} a_i^2 \delta_i^2}$

Then, for the resulting Cpk on a linear criteria $Y$ we obtain:

$$Cpk_Y = \frac{IT_Y}{6\sigma_Y} - \frac{\delta_Y}{3\sigma_Y}$$  \hspace{1cm} (13)

As demonstrated by Pillet, [P2], this Cpk$_Y$ presents a minimum value. A detailed calculation demonstrates that this minimal value is obtained when components biases are:

$$\left\{ \delta_i = \frac{2 S_Y}{a_i IT_Y}, i = I, N \right\} \quad \text{or} \quad \left\{ \delta_i = \frac{-2 S_Y}{a_i IT_Y}, i = I, N \right\}$$  \hspace{1cm} (14)

with $S_Y = \sqrt{\sum_{i=1}^{N} a_i^2 I_i^2}$

And this minimum value for Cpk$_Y$ is:

$$Cpk_Y_{\text{Min}} = \frac{IT_Y^2}{6 S_Y^2} \left( 1 - \frac{N}{9} \right)$$  \hspace{1cm} (15)

To guarantee a given Cpk target on a resulting criteria, that situation imposes a not easy calculation depending on $N$, the number of components, to verify we respect:

$$S_Y \leq \frac{IT_Y}{6 \sqrt{Cpk^2 + \frac{N}{9}}}$$  \hspace{1cm} (16)

This method is really effective to never more let mean biases to bring the disorder in an assembly, but, the constraint on components centering is heavy, and this method is mainly dedicated to adjustable processes.

To compare with usual statistical tolerancing method, we can consider a dimension usually specified by $A = 12 \pm 0.3$ for a Cpk$=1$, and becoming something as $B=12(0.1)$

Both specifications impose $\sigma \leq 0.1$, but for the first, nothing allows to refuse a batch with a Cpk$=1$ and a Cpk$=2$ (i.e. a mean of 12.15 and a standard deviation of 0.05).

With Inertial tolerancing, that batch will be never more accepted. A batch with a $Cp=1.33$ and a $Cp=2$ will be also rejected.

In addition to current practices, Inertial Tolerancing, impose a maximum mean bias (off-centering) of 1σ. This restriction allows to preserve the performance for the customer and brings a response to the non robustness of traditional statistical tolerancing, but for many industries, using heavy toolings that are not daily adjustable, this constraint is really not applicable. In fine blanking for example, we very often get very very small short term standard deviation nearby 1or 2μm, and that is really not the tolerance we can deal with for means variations resulting from initial tool fitting, wearing and maintenance along production, coils characteristics variations,…

The variability of parts inside a tooling can be very small, depending of its functioning, and in many case is totally independent to the parameters that affect the centering. In one case we talk about the accuracy of the tooling making the parts, in the second case we talk about the accuracy of all devices used to produce and maintain the tooling. This is the case for many other processes regular in serial production as molding, stamping, forging. According to this, Inertial Tolerancing is a nice idea for processes as machining, where it is possible to readjust many times a day, if needed, the production mean to the target, but is not flexible enough to be applicable in the general case.

Inertial Tolerancing has a demonstrated efficiency, but we think that to aggregate into a single parameter, the inertia, what influences the mean and what produces the dispersions is not the best way to pilot a production. As developed by Shewhart [SH1], a right statistical process control requires a set of two charts, one for the mean, and one for the dispersions.

As we are going to see, Process Tolerancing provide a solution to these problems (flexibility and SPC link)

### 3.6 Process tolerancing

This last method was proposed by W. Taylor in 1995 [T1], and is really close to the “Semi-Quadratic” method presented by B. Anselmetti [A1] & [A2].

To fix the problem of off-centering, the first idea is simply to specify a tolerance range for the production mean, then the specification on a component $X_i$ becomes:

- a target production target $t_i$
- a tolerance for the mean around this target $\pm \delta_i$
- a maximum standard deviation $\sigma_i$

The concept is to definitively decide to separate what concern the means and what concern the dispersions.

For that, W. Taylor suggests to specify the $X$ as:

$$X_i = t_i \pm \delta_i \pm 3\sigma_i$$  \hspace{1cm} (17)

with $t_i \pm \delta_i$ the tolerance interval for the mean $\mu_i$

and $\pm 3\sigma_i$ the max dispersion of parts around $\mu_i$

Srinivasan [SR1] propose to use a dual specification system, one giving the tolerance to apply on a single part, and a second dedicated to the SPC routines, and batch acceptance decisions by writing $\{ \sigma \leq \sigma_{\text{max}}, t-\delta \leq \mu \leq t+\delta \}$
The second idea of process tolerancing is to predict for a resulting criteria \( Y \):
- the resulting target \( t_Y \)
- the resulting \( Y \) mean tolerance range \( \pm \delta_Y \)
- and the resulting standard deviation \( \sigma_Y \)

For a linear criteria \( Y \) this calculation becomes easy:

\[
\begin{align*}
    t_Y &= a_Y + \sum_{i=1}^{n} a_i t_i \\
    \delta_Y &= \sum_{i=1}^{n} |a_i| \delta_i \\
    \sigma_Y &= \sqrt{\sum_{i=1}^{n} a_i^2 \sigma_i^2}
\end{align*}
\]

\( (18) \)

Figure 7: Typical result of process tolerancing

At specification stage, the off-centering \( \delta_Y \) is not a certainty, but only a risk, showing the potential range where we can get the mean. Sometimes a bias balances another...

At the opposite, \( \sigma_Y \) is something more definite, and is the expected short term standard deviation. By the way we get an accurate prediction on \( \text{Cp}_Y \).

For the calculation of the predictive \( \text{Cpk} \) on \( Y \), W. Taylor proposes to take the worst mean on each side and to calculate respectively \( \text{Cpl} \) and \( \text{Cpu} \).

Anselmetti [A2] suggests to introduce a probabilistic calculation for \( \delta_Y \) when the number of components is over or equal to 5, using an assumption of uniform distribution for the \( \delta_i \) and fitting the resulting distribution by a normal distribution for \( \delta_Y \). According to our experience, we think that there is a real economical interest to introduce a probabilistic approach also for \( N=2, 3, \) or 4. So many stack ups and systems are like this. Secondary the hypothesis of normal distribution for the convolution of 5 uniform distributions is quite not completely satisfying.

We propose here an innovative algorithm for the calculation of \( \delta_Y \) introducing a probabilistic approach working from \( N=1 \) to \( N=+\infty \) and using Beta distributions.

4 Proposal of an improvement for Process Tolerancing algorithm.

Anselmetti [A2] propose to introduce “a priori” uniform distribution for components mean biases. That is the best assumption we can do without data. Nevertheless, in many cases in automotive industry, we absolutely know that many tool makers, with a strong experience, are going to fit the mean not on the middle target, but on the side of maximum material to reduce the risk of scrap. The side of a potential rework is less risky for them. Some other tool makers, more professional, well know the wearing trend on their tool, and make usually the choice to offset the initial centering in the right side to increase the production time of good parts before tool changing or maintenance. These choices are not hazardous, but result to deterministic practices and demonstrate a high competence. In these cases, the uniform distribution is not appropriate.

A simple way to introduce this possibility of non-symmetry is to use Beta distributions

\[
\text{betapdf}(x) = \begin{cases} 
0 & \text{if } x < A \text{ or } x > B \\
\frac{1}{B(\alpha, \beta)} \left( \frac{x-A}{B-A} \right)^{\alpha-1} \left( \frac{B-x}{B-A} \right)^{\beta-1} & \text{if } A < x < B
\end{cases}
\]

with \( B(\alpha, \beta) = \frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha+\beta)} \)

\( (19) \)

Figure 9: Various shape of Beta distributions for \( A=0 \) and \( B=1 \)

Beta distribution is really flexible and offers the advantage to easily come back to uniform distributions \( (\alpha=\beta=1) \). On the figure above, the red curve is one option to represent an initial tool centering when a given reason exists to escape the
upper limit. Of course, to define “a posteriori” the right shape, the main problem stay the difficulty to get the data. Then, second step is to compute the resulting distribution for δY. The right and exact method, when Y is assumed as linear, is to use the convolution of distributions (see [SC1]). The theory is well known: if f and g are respectively the probability density functions of X1 and X2, the probability density function of Y=X1+X2 is:

\[ h(y) = \int_{-\infty}^{\infty} f(t) g(y-t) dt = f \otimes g \] (20)

Using FFT, the numerical computation is faster:

\[ h(y) = f \otimes g = \text{FFT}^{-1}(\text{FFT}(f) \times \text{FFT}(g)) \] (21)

This is here and exact method to get the distribution function for δY, after a primary step to get the probability density function of each variable (αi, δi). Then on resulting distribution, it is easy to determine the confidence interval for μY, for a given risk level. We propose to use a risk level near 0.1%. With a risk level at 0%, we come back to Worst Case calculation done by W. Taylor. This risk level is not directly linked to customer ppm, but represents the risk we accept to take to receive acceptable tools or batch but with an unlucky combination producing bad assemblies.

The calculation is easy with tools as Scilab, Mathematica, or Matlab, but to simplify the computation and to program it in tools as Excel or equivalent, we propose a simplified but effective method. The idea is simply to fit the resulting distribution by a Beta distribution. There are many reasons to justify this choice:
1. We are right to wait for a bounded distribution for δY according to that all distributions of δi are bounded.
2. Beta distribution, with its 4 parameters, is really flexible.
3. From the Central Limit theorem, we know that when the number of components increase, the resulting distribution converge to a normal distribution, and beta distribution converge very well to a normal distribution when α=β→+∞ (with α=β=10, the result is really close to normality)
4. for n=1, the result fully is exact.
5. the convolution of 3 uniform distributions is very close to a beta distribution,

The computation becomes very quick and easy, after defining the distribution for each δi, by identifying the parameters (αi, Bi, αi, βi) , we search the parameters (A, B, α, β) to fit the distribution of δY.

Step 1 : we calculate the mean and the variance of each δi

\[ \overline{\delta_i} = \frac{\alpha_i B_i + \beta_i A_i}{\alpha_i + \beta_i} \] (22)

\[ V_i = \frac{\alpha_i \beta_i (B_i - A_i)^2}{(\alpha_i + \beta_i)^2 (1 + \alpha_i + \beta_i)} \] (23)

Step 2 : we calculate the mean and the variance of δY.

\[ m = \overline{\delta_Y} = \sum_{i=1}^{n} a_i \overline{\delta_i} \quad \text{and} \quad V = Var(\overline{\delta_Y}) = \sum_{i=1}^{n} a_i^2 V_i \]

Step 3 : we determine Ay and By :

\[ \Delta = \sum_{i=1}^{n} |a_i| (B_i - A_i) \rightarrow A_Y = -\frac{\Delta}{2} \quad \text{and} \quad B_Y = +\frac{\Delta}{2} \] (24)

Step 4 : we determine αY and βY

\[ \alpha_Y = \frac{(B_Y - m) (m - A_Y)}{\Delta V} \] (25)

\[ \beta_Y = \frac{(m - A_Y) (B_Y - m)}{\Delta V} \] (26)

Now, it is easy to determine the value for δY at a given risk level p (we suggest p=0.1%) 

\[ \delta_Y (p%) = \text{Betainv}(1 - p%, \alpha_Y, \beta_Y, A_Y, B_Y) \] (27)

We propose to use this value in place the value calculated with the Worst Case routine used by W. Taylor.

Finally, for the Y, the calculation of Cpl and Cpu at the unilateral risk level p% is :

\[ Cpl_Y = \frac{t_Y - \delta_Y - Lsl}{3 \cdot \sigma_Y} \quad \text{and} \quad Cpu_Y = \frac{Usl - t_Y - \delta_Y}{3 \cdot \sigma_Y} \] (28)

The main interest to this probabilistic approach is to be able to calculate the risks to not achieve the required Cpl and Cpu targets on the Y, respectively rL and rU :

\[ rL = \text{betacdf} \left( Lsl - t_Y - 3 \cdot \text{Cpl} \cdot \sigma_Y, \alpha_Y, \beta_Y, A_Y, B_Y \right) \] (29)

\[ rU = 1 - \text{betacdf} \left( Usl - t_Y + 3 \cdot \text{Cpl} \cdot \sigma_Y, \alpha_Y, \beta_Y, A_Y, B_Y \right) \] (30)

Another improvement for tolerancing optimization is to be able to calculate the optimal target for Y, \( ty_{\text{opt}} \), and the maximal associated standard deviation \( \sigma_{y_{\text{opt}}} \) for a given bias risk on Y \( \delta_Y \).

\[ ty_{\text{opt}} = \frac{(Usl - \delta_Y) \times \text{Cpl} + (Lsk + \delta_Y) \times \text{Cpu}}{\text{Cpl} + \text{Cpu}} \] (31)

\[ \sigma_{y_{\text{opt}}} = \frac{Usl - Lsl - 2 \delta_Y}{3 \cdot (\text{Cpl} + \text{Cpu})} \] (32)
5 Proposal of SPC tools and routines adaptations for Process Tolerancing.

5.1 Tools for X quality management

Using Process tolerancing, each X (component or process parameter) must now fulfills two independent requirements:

- \( t - \delta \leq \mu_{\text{prod}} \leq t + \delta \)
- \( \sigma_{\text{prod}} \leq \sigma \)

We never more talk about Cpk for an X. Cpk, Cpl and Cpu are only relevant and applicable for a resulting customer criterion (the Y). Then, it becomes necessary to adapt quality tools to manage and control the production on X parameter.

The tool we suggest, named XQA, is somewhat as a third 3D view of the \([\text{Xbar}, \text{S}]\) charts from Shewhart [SH1], on which one we propose to integrate from an innovative way the sampling uncertainty.

Generally in the industry, we never know in advance the production mean or its standard deviation. From a sample of N parts, we only are able to determine confidence intervals.

For the standard deviation, when the production mean is unknown we use the usual theorem:

\[
\frac{(N-1)s_N^2}{\sigma^2} \sim \chi^2(N-1)
\]

With \( \sigma^2 \) the unknown production variance

\[
s_N^2 = \frac{1}{N-1} \sum_{i=1}^{N} (x_i - \bar{x})^2
\]

the unbiased variance estimator and \( \bar{x} \) the sample mean.

Then \( \sigma_p \) confidence interval for a unilateral risk of \( \alpha \) is:

\[
\sqrt{(N-1)s_N^2} \leq \sigma_p \leq \sqrt{X_{\chi^2}(N-1)}
\]

The second result to be used here is the following:

\[
\frac{\bar{x} - \mu_p}{s_N/N} \sim \text{Student T distribution for } \nu = N - 1
\]

And we get the confidence interval the production mean \( \mu_p \):

\[
\bar{x} - T_{\alpha,N-1} \times \frac{s_N}{\sqrt{N}} \leq \mu_p \leq \bar{x} + T_{\alpha,N-1} \times \frac{s_N}{\sqrt{N}}
\]

It becomes possible to represent the confidence domain for the couple \((\mu_p, \sigma_p)\) around the sample point (xbar, sN).

The usual cut-off ellipses of bi-normal distributions become here something as an egg contour (resulting from transformations through CDF and CDF_inverse functions). This representation allows to visualize graphically the uncertainty coming from the sampling hazard, without to need to show the formulas (33) and (34).

When the egg is fully in the green, we preserve the customer with a risk under the selected level \( \alpha \% \). When the egg touches the specification borders, we have to manage an orange alert (control of more parts, cascading the orange alert on the impacted Ys…). When the egg is fully out the green zone, the chance for the supplier is less than the selected level \( \alpha \% \), and we manage a stop (red alert, quality wall, reject, rework…).

In reality, the resulting risk \( \alpha \% \) requires a more sophisticated calculation. We first have to determine the left ad right side risks with the student distribution, and then the up risk with the Chi2 distribution.

\[
\alpha_L = T_{\text{cdf}} \left( \frac{1 - \delta - \bar{x}}{s_N/N}, N - 1 \right)
\]

\[
\alpha_R = 1 - T_{\text{cdf}} \left( \frac{t + \delta - \bar{x}}{s_N/N}, N - 1 \right)
\]

\[
\alpha_U = 1 - \text{Chi2cdf} \left( N - 1 \times \left( \frac{s_N}{\sigma_{\text{max}}} \right), N - 1 \right)
\]

Finally we get:

\[
\alpha = \alpha_L + \alpha_R + (1 - \alpha_L - \alpha_R) \cdot \alpha_U
\]

When a balloon touch a border, and becomes orange, the risk to be not ok is so high to preserve the customer, but also so small to stop the producer to hope to be ok. No decision can be taken with a risk under the desired level. In the goal to solve this indecision it is possible to reverse the calculation (formulas (35) to (38)) and to estimate the number of parts to be measured to take the decision with the desired risk level.

5.2 SPC control chart adaptation for X

The graph XQA (figure 11) is a view without history where we check the acceptance batch after batch. If we want to pilot
a criterion along time, it becomes necessary to adapt the usual Xbar control chart. The following figure shows a way to survey the centering of the mean integrating graphically the sampling risk. Here the height of each box is calculated as the width of XQA balloon (egg contour). Of course, the same kind of adaption can be done for the usual S chart.

With this boxes representation, it is possible to integrate in a single chart a set of samples with different sizes.

**Figure 13 : Xbar chart adaptation for Process Tolerancing**

5.3 Extension for Y quality assessment

We propose also to extend this kind of representation of sampling risk for the customer resulting criteria, the Ys. The following figure shows the proposed graph for a Y with a specification range [Usl, Lsl] and different Cpk target on each side (Cpl ≠ Cpu).

**Figure 14 : Y Quality Assessment**

With:

\[
 t_Y = \frac{Usl.Cpl + Lsl.Cpu}{Cpl + Cpu}
\]

(39)

And:

\[
 \sigma_{Y_{\text{Max}}} = \frac{Usl-Lsl}{3.(Cpl + Cpu)}
\]

(40)

In this representation, the triangle of acceptance stays the same whatever the size of the sample. That is here a consistent difference with the representation proposed in the norm NFX06-034 (Appendix B).

6 Conclusion

Built on the same argumentation, Inertial Tolerancing and Process Tolerancing offer two solutions to bring the required robustness to statistical tolerancing. At first, Inertial Tolerancing seems to be simpler, the tolerance can be expressed using a target and a single other number, the inertia. However, by aggregating what talks about the mean and what regards the dispersions, and with its resulting constraint on mean centering (δ ≤ I ) inertial tolerancing is not appropriate to industries using heavy tools that are not daily adjustable. For many processes, tool centering, and then mean bias, is fully independent to the standard deviation. From this point of view, Process Tolerancing, including our improvements offers a larger application domain. The possibility to specify separately mean centering, according to tool fitting possibilities, including maintenance exigencies, and dispersions according to tool working, offers a real improvement for automotive industry. Nevertheless, whatever the selected method, the management of practices changing, from design office to the factory, stays the most difficult task.

7 References


