SHIFT, NARROW, AND CHOP TO IMPROVE PROCESS CAPABILITY

Alan Bowman
Josef Schmee

Union Graduate College
80 Nott Terrace
Schenectady, NY, 12308, USA

ABSTRACT

When output random variables are a function (known as a transfer function) of input random variables, Monte Carlo simulation has often been used to examine the sensitivity of the outputs to changes to the inputs. An important and commonly used measure of the outputs is their process capability (the probability that an output is within specification limits). In this paper, we show how to efficiently conduct extensive analysis of the sensitivity of the process capability of outputs to changes to inputs. Specifically, we show how a single set of simulation replications can be used to efficiently estimate the process capability as a function of each input random variable's values, its parameters, and truncation of its values at chosen limits. The approach is extremely flexible; the effects of changes to the distributional form of an input variable alone or in combination with the previously mentioned changes are easily evaluated.

1   INTRODUCTION

A fundamental paradigm of many quality improvement approaches is to make improvements to carefully selected X variables in order to reduce the likelihood that one or more critical-to-quality Y variables will fall outside acceptable specification limits. An important class of such applications is where the Y variables are modeled as transfer functions of the X variables. A function g is referred to as a transfer function if $Y = g(\mathbf{X}) = g(X_1, X_2,\ldots, X_n)$, where the vector of input variables $\mathbf{X}=(X_1, X_2,\ldots, X_n)$ is transformed into the output variable Y. The output variable Y is a random variable if at least some of the X_i inputs are random variables (which is frequently the case in practice and is the case we treat in this paper). The term transfer function conveys the fact that the variability in the X_i inputs is transferred to the output Y. This transfer is, of course, a major reason that Y values fall outside the specification limits (these values are said to be defective). To avoid defectives, one can try to choose target values for the X_i's (**shift** the means of the random variables) that are robust to the variability transfer, reduce the variability in the X_i's (**narrow** the probability distribution) that is getting transferred, or limit the ranges of the X_i's (similar to reducing the variability but here we refer to truncating or **chopping** the distribution rather than narrowing it) as the transfer functions often convert extreme X_i values into defective Y values. These shift, narrow, and chop efforts aimed at avoiding defective Y values are the focus of this paper.

We present a series of tools that utilize key information from Monte Carlo simulation (MCS) of a transfer function to allow potential shift, narrow, and chop actions to be efficiently evaluated in a unified framework. The tools involve both computational and graphical features. Common to all the tools presented here is that they require only one set of simulation replications – any or all can be utilized without making any changes to the simulation model and running additional simulations. Specifically, we develop the following abilities:

1.   The ability to efficiently generate a curve showing the probability that Y will be defective as a function of any X_i's values.

2. The ability to examine the effect of any change to the probability distribution of any X_i variable on the probability that Y will be defective. Allowable changes to an X_i variable include changes to one or more parameters of the probability distribution, changes to the distributional form, and truncations of the distribution. In essence, the effects of a completely different distribution for an X_i on the probability that Y will be defective can be evaluated without re-simulating.

3. The ability to generate a curve showing the probability that Y will be defective as a function of any parameter of any X_i's probability distribution.

As we will describe, our approach hinges on the ability to solve for values of an X_i variable that, given specific values for all other input variables X_j, $j \neq i$, will yield values equal to the specification limit values for Y. Although this can be done for essentially all transfer functions, it is more efficient to do so when closed-form solutions can be obtained to make these computations. We believe this is possible for most transfer functions in practice but not all. For example, this would be true for transfer functions that are linear, quadratic, or cubic in nature with unlimited cross-products.

The approach can be applied when there are multiple Y variables and/or the transfer function(s) are not monotonic; the easiest case to describe is when there is a single Y variable and its transfer function is monotonic. For exposition clarity and brevity, this paper will focus on how to handle this case and will demonstrate the approach on practical examples. If there are multiple Y variables but it is desired to analyze their defective probabilities individually and then combine them (for example, with a weighted combination based on relative importance), the extension is trivial. In that case, the approach we will describe can simply be applied to each Y variable in series. With multiple Y variables, however, it would frequently be useful to examine the probability that none of the Y variables are defective. In this case, or the case where a transfer function is non-monotonic, or the case where both of these complicating features are present, the approach is very similar and equally efficient to apply but requires more detailed bookkeeping. Because of this bookkeeping, this approach requires extra notation and is more complex and lengthy to describe. Details are provided in the working paper by Bowman and Schmee (2011).

We assume that each of the random X_i's is modeled by a continuous probability distribution (possibly degenerate) and that they are independent. All of the approaches we present (both graphical and computational) allow for sensitivity analysis to be done for one X_i variable at a time. We believe that the ability to incorporate correlated X_i's and/or the ability to analyze simultaneous changes to multiple X_i's would be important extensions.

In Section 2, we review the literature and place the contribution of this paper in that context. In Section 3, we describe the simulation steps that are common to all 3 abilities that we have listed above. In Section 4, we show how to use the information obtained from these steps to efficiently evaluate the probability that a Y variable is defective. An example is used for illustration. Although this is not one of the 3 abilities we are focused on in this paper, it is a useful ability in and of itself; more importantly, it provides the fundamental building block for the main 3 abilities we describe. In section 5, we develop the first ability described above. In section 6, we develop the 2nd and 3rd abilities, which are closely related. Finally, in Section 7, we offer conclusions and recommendations for future research.

2   LITERATURE REVIEW

Tolerance design is a research topic that has received much attention; it involves examining the effects of component tolerances on performance variables and incorporating the results into the system design. A thorough review of this field is well beyond the scope and space limitations of this paper. The interested reader is primarily referred to two good overviews. Evans (1975) gives a broad overview of the statistical problems of tolerance design. More recently, Creveling (1997) presents a comprehensive approach (and review of the literature), using experimental design, Taguchi methods, and sensitivity analysis that includes the use of MCS. Tolerance design is a special focus of the Six Sigma approach as described by Snee and Hoerl (2003), where input X-variables are modified to yield desired output Y-variable values. The Six Sigma approach often uses transfer functions to model the relationship between these inputs and

outputs. It implies that in order to improve quality one must ultimately control the X-variables (see also Kumar et al. (2008)).

Another related topic is capability analysis, which utilizes several metrics to assess variables of interest for their capability to meet specifications. Kotz and Johnson (1993), Spiring et al. (2003), and Wu, Pearn, and Kotz (2009) give overviews of various capability indices. Capability indices are typically subject to distributional assumptions. To avoid this problem, Polansky (2001) has used a nonparametric approach. Our paper, instead of using a traditional capability index, uses P(D), the probability of a defective. This approach has been previously used by Wu and Hamada (2000) and Chiao and Hamada (2001) who consider optimizing probability of defectives in a multiple response setting based on experimental designs. Chiao and Hamada (2001) maximize the probability that all responses meet their respective specifications simultaneously.

The response surface is the most widely used transfer function and is, therefore, another important research area that is directly relevant to this paper. Myers, Khuri, and Carter (1989), Myers and Montgomery (1995), Myers et al. (2004) and Myers Montgomery, and Anderson-Cook (2009) provide in depth reviews of many issues surrounding response surfaces. Carlyle, Montgomery, and Runger (2000) and Myers et al. (2004) review multiple response optimization approaches. In many of these approaches, the error terms for the Y's are explicitly or implicitly (for example, when capability indices are used) assumed to be normally distributed. Bowman, Doganaksoy, and Schmee (2010) show an efficient optimization of a multi-response transfer function without the need to make distributional assumptions by utilizing gradient estimates obtained from the approach described by Bowman and Schmee (2004).

Increasingly, the complexity of tolerancing, capability index, and response surface issues has led to the use of Monte Carlo simulation. For example, Boning and Mozumder (1994) used it to design semiconductors, while Kalil, Maugeri, and Rodrigues (2000) used simulation of response surfaces to design and optimize an alcoholic fermentation process. The topic has also attracted methodological discussions such as in Skowronski and Turner (1997).

We are now in a position to relate our terms of shift, narrow, and chop to the areas of statistical tolerancing, capability indices, and response surface optimization and to clarify the contribution of this paper. Statistical tolerancing has dealt with both narrow and chop elements with more attention being placed on narrow than chop. Response surface optimization associated with quality improvement has focused primarily on the shift element with some treatment of narrow. Capability indices employ metrics other than the probability of defectives and often involve distributional assumptions. In addition to allowing complete freedom in modeling inputs and making no distributional assumptions about outputs, our paper has two primary distinguishing characteristics. The first is that it treats all three elements (shift, narrow, and chop) in a unified framework. By this, we mean that the methodological approach is the same for all three elements and also that all three elements can be evaluated simultaneously (along with any interaction effects) rather than separately and independently.

To understand the second distinguishing characteristic, it is first necessary to briefly consider the methodology that we employ. Since at least some of the X_i's are random variables and analytical results for transfer functions of random variables with general distributions are limited, transfer functions are often analyzed using Monte Carlo simulation. In this context, key to the ability to avoid defective Y values is the ability to use MCS to assess the sensitivity of the probability of the Y's being defective to changes in the X_i's. This sensitivity analysis in the past has looked at infinitesimal changes to the X_i's or the parameters of their probability distributions and the results have often been utilized in optimization schemes. The second distinguishing characteristic of this paper, then, is that the tools we present enable the effects of significant (rather than infinitesimal) changes to X_i's or their probability distributions on the probability of defectives to be clearly and efficiently assessed. As these changes are significant and can involve shift, narrow, and chop effects simultaneously, the primary use of these tools would be to efficiently evaluate practical alternatives and directly focus improvements on the alternatives with the biggest impact; they are not primarily useful as information to be embedded in an optimization scheme.

The software package JMP, version 8 (JMP 2009) or higher, offers a profiling method based on MCS that can be used to study problems similar to the ones discussed in this paper and, in particular, to accomplish the first ability that we described in the introduction. A key difference is that JMP re-simulates at selected values of input variables and uses interpolation, whereas the method proposed in this paper is an efficient one-pass approach that evaluates an entire range of values of input variables based on one set of simulations. As we will describe, this requires the ability to gather specific information from the simulation runs and this can be done efficiently when the transfer functions can be inverted to express an input variable as a closed form function of each output variable and the other input variables. We believe this to be possible for most (but certainly not all) transfer functions in practice. When it is not possible, we recommend using the JMP approach to accomplish the first ability outlined in the introduction and the approach we describe to accomplish the second and third abilities.

3   CORE SIMULATION STEPS

All the tools that we present are simulation-based. They begin by simulating the transfer function a number of times and identifying the same key information from each run. This information is then tracked and manipulated to efficiently accomplish the various abilities. Each individual simulation run requires the following:

1. Simulate X_1, X_2,…, X_n from their input distributions.
2. For each X_i, determine which range(s) of values X_i could take on so that Y does not violate its specification limits, while holding the other X_i's constant at their simulated values. In monotone relationships, this produces a single range of values between what we call $X_{min(i)}$ and $X_{max(i)}$. Thus, for each simulation run, we identify $X_{min(i)}$ and $X_{max(i)}$ such that Y is good if and only if $X_{min(i)} \leq X_i \leq X_{max(i)}$. This is done by solving for the X_i values that, if plugged into the transfer function along with the simulated values for all other X_j's, j ≠ i, yield values equal to the specification limit values for the Y variable.

A simple example demonstrates the calculations of $X_{min(i)}$ and $X_{max(i)}$. Suppose a laminate is formed from two layers. The total thickness $Y = X_1 + X_2$, where X_1 and X_2 are the thicknesses of the two layers. The lower and upper specification limits on Y are $Y_L = 2.95$ and $Y_U = 3.05$, respectively. (The units of the thicknesses are omitted.) X_1 and X_2 are normally distributed random variables with means of 1 and 2 and standard deviations of 0.01 and 0.015 respectively. Suppose a single simulation run results in $x_1 = 0.99$ and $x_2 = 1.98$. The total (simulated) thickness y=0.99+1.98 = 2.97 is within specification limits. The calculations for $X_{min(1)}$ and $X_{max(1)}$ assume that x_2 is fixed at 1.98:

$X_{min(1)} = Y_L$-$x_2 = 2.95 - 1.98 = 0.97$.
$X_{max(1)} = Y_U$-$x_2 = 3.05 - 1.98 = 1.07$.

$X_{min(1)}$ and $X_{max(1)}$ can be interpreted to mean that for x_2=1.98, Y will be within its specification limits as long as X_1 is in the interval from 0.97 to 1.07.
The calculations for $X_{min(2)}$ and $X_{max(2)}$ assume that x_1 is fixed at 0.99:

$X_{min(2)} = Y_L$-$x_1 = 2.95 - 0.99 = 1.96$.
$X_{max(2)} = Y_U$-$x_1 = 3.05 - 0.99 = 2.06$.

$X_{min(2)}$ and $X_{max(2)}$ can be interpreted to mean that for x_1=0.99, Y will be within its specification limits as long as X_2 is in the interval from 1.96 to 2.06.

This simple example has two-sided specification limits. Some applications have one-sided specification limits, but this is tantamount to setting either $Y_L = -\infty$ or $Y_U = +\infty$. Correspondingly, either $X_{min(i)}$ or $X_{max(i)}$ could be $-\infty$ or $+\infty$, so that the range could be one-sided, two-sided, or even empty.

4   EFFICIENT ESTIMATION OF DEFECTIVE PROBABILITIES

When estimating the probability of defectives using simulation, the straightforward procedure is simply to divide the number of replications for which there was a defective by the total number of replications. The standard deviation of this estimate would be $\sqrt{p(1-p)/n}$. This is quite large (compared to the mean)

for systems that are highly capable unless the number of replications is very large. For example, if p=.00001, it would take 1 million replications just to get the standard deviation to equal the mean. A more efficient approach is to take advantage of the fact that, holding all the other X_j, $j{\neq}i$, constant at their simulated values for a simulation replication, the $X_{min(i)}$ and $X_{max(i)}$ values define the interval for X_j on which the Y results in a good outcome. Let $P_i(D \mid$ fix all X_j, $j{\neq}i)$ refer to the probability that Y is outside its specification limits, holding fixed all X_j, $j{\neq}i$ and let $F_{Xi}(x) = P(X_i \leq x)$. We have:

$$P_i(D \mid \text{fix all } X_j, j{\neq}i) = P(X_i < X_{min(i)}) + P(X_i > X_{max(i)}) = F_{Xi}(X_{min(i)})) + [1 - F_{Xi}(X_{max(i)})]. \quad (1)$$

Using (1), one can calculate a value of $P_i(D \mid$ fix all X_j, $j{\neq}i)$ for each run. It is an estimate of P(D), conditional on the values X_k, $k{\neq}i$, for the simulation replication. These conditional estimates could be averaged across all simulation replications to yield an estimate of P(D). This estimator would be unbiased (as E(E(Y|X))=E(Y) for any Y and X). Compared to the straightforward estimator, however, it would have lower variance (as Var(E(Y|X))=Var(Y)-E(Var(Y|X))). This idea and formula are at the heart of all variance reduction approaches using conditional estimates within MCS (see, for example, Law and Kelton (1991)). They are also at the heart of the approaches to all 3 abilities that we will develop.

We note that one could follow the same approach for each X_i (fixing X_j, $j{\neq}i$ at their simulated values) to obtain an estimate of P(D), similarly conditional on the values X_j, $j{\neq}i$, for the simulation replication and average these conditional estimates across all simulation replications to yield an estimate of P(D). In this manner, an estimate of P(D) is available from each input variable. To be clear, we refer to the estimate obtained for P(D) working with the $X_{min(i)}$ and $X_{max(i)}$ values each replication as $P_i(D)$. One could use any $P_i(D)$ as the estimate of P(D) or any weighted combination thereof. We demonstrate this approach using a small example.

Bank Processing Example: This example concerns the bank processing time of loan approvals. The approval process consists of four sequential steps: receiving application, review of application, credit check of applicant, and loan finalization. In this example we assume that each step follows a normal distribution: $X_1{\sim}N(13,1)$, $X_2{\sim}N(14,2)$, $X_3{\sim}N(15,3)$, $X_4{\sim}N(16,4)$, where $X_i{\sim}N(\mu_i,\sigma_i)$ means that the time X_i to complete step i is normally distributed with mean μ_i and standard deviation σ_i. The parameters are given in hours. We choose the normal distribution because it affords us a simple analytic solution with which to compare the simulation-based derivative estimates. The $Y= g(X_1, X_2, X_3, X_4) = X_1+X_2+X_3+X_4$.

Let's suppose that Y has a lower specification limit of 50 and an upper specification limit of 70. Suppose the simulated values for a particular simulation were:

x_1	x_2	x_3	x_4
12.957	14.631	15.004	14.631

Since the sum of these values is 57.223, Y falls within the specification limits and a straightforward simulation would simply record the fact that Y was good for this particular run. Alternatively, however, we could calculate the $X_{min(i)}$ and $X_{max(i)}$ values for any or all of the X_i's. For example, if we do this for i=4, we would have $X_{min(4)}$ = 50 – (12.957+14.631+15.004) = 7.408. Similarly, $X_{max(4)}$ = 70 – (12.957+14.631+15.004) = 27.408. Applying (1), we compute $P(X_4 < 7.408) + P(X_4 > 27.408) = .0159 + .0022 = .0181$. Thus, we have computed $P_4(D \mid$ fix all X_j, $j{\neq}4) = .0181$. Similarly, we calculated $X_{min(i)}$ and $X_{max(i)}$ for i = 1, 2, and 3 and used those values to compute $P_1(D \mid$ fix all X_j, $j{\neq}1) = .0000$, $P_2(D \mid$ fix all X_j, $j{\neq}2) = .0005$, and $P_3(D \mid$ fix all X_j, $j{\neq}3) = .0081$.

We conducted 10,000 simulation replications using this example. In these 10,000 replications, Y was defective 849 times so that the standard simulation estimate would be P(D) = .0849 (the analytic solution is .0863). Using our 10,000 replications, we computed $P_1(D) = .08591$, $P_2(D) = .08589$, $P_3(D) = .08595$, and $P_4(D) = .08533$.

As we mentioned, any weighted combination of the $P_i(D)$'s would be also be unbiased. Getting the lowest variance combination would be a useful extension but is not the focus of this paper. Instead, we will focus on using the idea behind these estimators to build the 3 abilities we have described. We observe, however, that if an X_i tends to fall either completely within the range between $X_{min(i)}$ and $X_{max(i)}$ or completely outside that range then the variance mimics the binomial variance. Hence, X_i's that have a large amount of variance tend to lead to lower variance estimates. For example, for our problem, the standard error of $P_1(D)$ was .00247 for i=1, .00209 for i=2, .00165 for i=3, and .00114 for i=4. (Note that any of these would be an improvement over the standard error of the standard simulation estimate, which would be .00279).

In the cases where there are multiple Y variables and the measure of interest is the probability that none are defective, or any of the transfer functions are non-monotonic, or both, there will not be a unique $X_{min(i)}$ and $X_{max(i)}$ for each X_i . As mentioned, the extension to handle this situation is easy and efficient to apply but more complex and lengthy to describe and requires extra notation to be introduced. It is for this reason that we refer the reader to the working paper by Bowman and Schmee (2011) for the details.

5 DEFECTIVE PROBABILITIES AS A FUNCTION OF AN X_I'S VALUES

Suppose one wanted to estimate the probability that Y is defective given that an X_i is fixed at a value x_i. The procedure for doing this is a logical extension of the previous section. In addition, the ability to answer this question will be extended in the next section to the ability to evaluate the effect of any change or set of changes (shift, narrow, chop, even a complete change to the distributional form) to an X_i's distribution on the probability that Y is defective.

The traditional simulation method to estimate the probability that Y is defective given that an X_i is fixed at a value x_i, $P(D \mid X_i = x_i)$, would be as follows. For each replication, enter the value x_i together with the other simulated X_j, $j \neq i$, values into the transfer function. Then determine whether the outcome Y was good or defective, and calculate the fraction of replications for which the Y was defective. A second way to get the same answer is to check whether x_i is below $X_{min(i)}$ or above $X_{max(i)}$ for each replication.

Since it is of interest to perform these checks for any fixed value x_i, in the range of X_i, we could essentially follow this second approach. However, by using the simulation replications to build an empirical distribution function for $X_{min(i)}$, call it $F_{Xmin(i)}(x_i)$, and for $X_{max(i)}$, call it $F_{Xmax(i)}(x_i)$, one does not have to make this check for each simulation run. Instead, at the end of all the simulation replications, one can estimate $P(D \mid X_i = x_i)$ as follows:

$$P(D \mid X_i = x_i) = P(X_{min(i)} > x_i) + P(X_{max(i)} < x_i) = [1 - F_{Xmin(i)}(x_i)] + F_{Xmax(i)}(x_i). \qquad (2)$$

Note that (1) and (2) are very similar. The only difference is that (1) treats $X_{min(i)}$ and $X_{max(i)}$ as constants fixed at their values calculated for a simulation replication and X_i as a random variable. Equation (2) treats X_i as a constant fixed at x_i and $X_{min(i)}$ and $X_{max(i)}$ as random variables.

The advantage of (2) is that, once the distribution functions for $X_{min(i)}$ and $X_{max(i)}$ are estimated, one can use them to compute $P(D|X_i = x_i)$ for any value x_i. One can do this because $X_{min(i)}$ and $X_{max(i)}$ do not depend on X_i, but only depend on the values of the other input variables X_j, $j \neq i$. Based on a single set of simulation replications, one could thus produce $P(D|X_i = x_i)$ as a function of x_i. In addition, following the same procedure for each of the other X_j's allows $P(D|X_j = x_j)$ as a function of x_j to be produced, all using the same set of simulation replications.

Bank Processing Example: These concepts can be demonstrated with the Bank Processing Time example of loan approvals. The specification limits used in this demonstration are $Y_L = 42$ and $Y_U = 76$. The limits have been chosen fairly wide so that the system becomes fairly capable. They are asymmetric around the mean of 58.

We demonstrate the concepts on the X_1-variable where $X_1 \sim N(13,1)$. Figure 1 shows the empirical CDF's of $X_{min(1)}$ and $X_{max(1)}$ obtained from 10000 simulation replications.
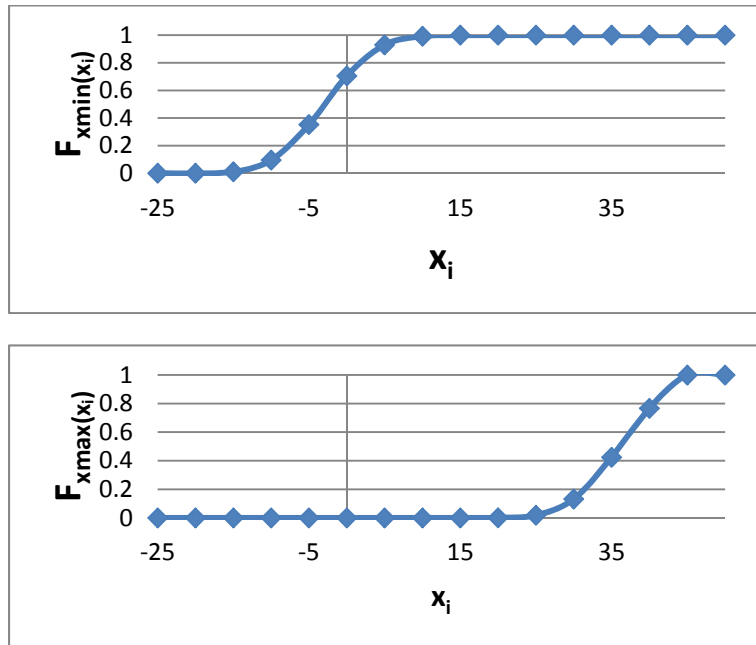
Figure 1: Empirical CDF's of $X_{min(1)}$ and $X_{max(1)}$

As expected, because all input distributions are normal and the output variable is a sum of these normals, $X_{min(1)}$ and $X_{max(1)}$ are also normally distributed.

The empirical CDF's of $X_{min(1)}$ and $X_{max(1)}$ are then used to calculate $P(D \mid X_1 = x_1)$ using Equation (2). The calculations are demonstrated in Table 1. Column 2 is labeled $P(D|x_1$ for $X_{min(1)})$ and contains empirical estimates of $P(X_{min(1)} > x_1) = [1 - F_{Xmin(1)}(x_1)]$. Column 3 is labeled $P(D| x_1$ for $X_{max(1)})$ and contains $P(X_{max(1)} < x_1) = F_{Xmax(1)}(x_1)$. Column 4 is the empirical estimate of the probability defined in Equation (2), i.e., $P(D \mid X_1 = x_1) = P(X_{min(1)} > x_1) + P(X_{max(1)} < x_1) = [1 - F_{Xmin(1)}(x_1)] + F_{Xmax(1)}(x_1)$.

Table 1: $P(D \mid X_1 = x_1)$

x_1	$P(D|x_1$ for $X_{min(1)})$	$P(D|x_1$ for $X_{max(1)})$	$P(D|x_1)$
9	.0137	.0000	.0137
9.5	.0104	.0000	.0104
10	.0077	.0000	.0077
10.5	.0062	.0000	.0062
11	.0045	.0000	.0045
11.5	.0037	.0000	.0037
12	.0025	.0000	.0025
12.5	.0021	.0000	.0021
13	.0016	.0000	.0016
13.5	.0009	.0001	.0010
14	.0007	.0002	.0009
14.5	.0004	.0004	.0008
15	.0003	.0008	.0011
15.5	.0002	.0010	.0012
16	.0002	.0013	.0015
16.5	.0001	.0020	.0021
17	.0000	.0030	.0030

A graph of column 4, then, is a graph of P(D | $X_1 = x_1$) as a function of x_1. In exactly, this manner, such a graph could be produced for each X_i. These graphs are extremely interesting and should be invaluable for conducting sensitivity analysis. For example, one key question is where the minimum would occur. The example graph (Figure 2) shows that the empirical minimum occurs near 14. This is as expected for this simple example because it would move the average of the sum to exactly half way between the upper and lower specification limits on Y. For more complex transfer functions and other probability distributions, however, the answer would not be obvious.
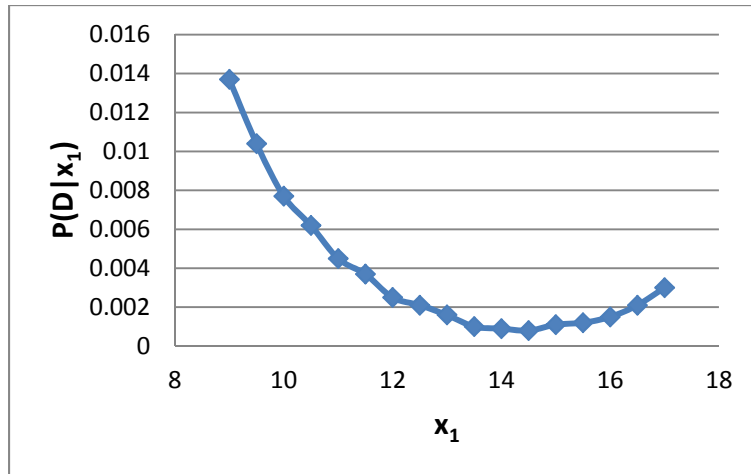

Figure 2: P(D | $X_1 = x_1$)

6   THE EFFECT OF CHANGES TO THE PROBABILITY DISTRIBUTION OF AN X ON THE PROBABILITY OF Y BEING DEFECTIVE

In the previous section, we described how to use a single set of simulation replications to estimate the function P(D | $X_i = x_i$) for every X_i. Since $\int g(x) f(x) dx = E[g(X)]$, where $f_{X_i}(x_i)$ is the probability density function of the random variable X, we could approximate $\int P(D|X_i = x_i) f_{X_i}(x_i) dx$ using our estimated function P(D| $X_i = x_i$) and produce an estimate of P(D).

Following the approach of the previous section, we could estimate P(D|$X_i = x_i$) at selected discrete values for x_i over a desired range. Once this is done, we could approximate the above integral as a standardized sum:

$$P_{X_i}(D) = \frac{\sum_{i=1}^{n} P(D|X_i = x_i) f_{X_i}(x_i) dx}{\sum f_{X_i}(x_i) dx} \qquad (3)$$

Here, j is the index for the x_i values and there are n values used (j=1 to n). Note that if the P(D | $X_i = x_i$) function is estimated at uniformly spaced values (as it typically would be), the dx would drop out of the calculation.

The probability density $f_{X_i}(x)$ is the most important concept here. As it is applied at this stage, $f_{X_i}(x)$ can be any density, not just the one originally used in the simulation. In applications one would choose $f_{X_i}(x)$ different from the originally simulated distribution for at least three important reasons:

1. Shift: To evaluate the impact of changing the mean of the originally simulated distribution. In the Bank Processing Example one may want to know the effect on P(D) from a change in the X_1 mean from 13 to 12.
2. Narrow: To evaluate the impact of changing the standard deviation of the originally simulated distribution. In the Bank Processing Example one may want to know the effect on P(D) from a change in the X_1 standard deviation from 1 to 0.5.
3. Chop: To evaluate the impact of truncating the tails of the distribution by imposing tolerance limits on X_i. In the Bank Processing Example one may want to know the effect on P(D) from a change to the range of admitted X_1 values by truncating values outside the tolerance limits.

Each of these problems can be dealt with by the same approach. In fact, the same approach can handle any combination of changes to any individual X's distribution, including a complete change of distributional form.

Bank Processing Example Revisited: In Table 2, we show how to evaluate truncation (chop) for X_1 of the Bank Processing example. For expositional purposes, we use rather wide increments ($\Delta x=0.5$) for calculating $f_{X_i}(x)dx \approx f_{X_i}(x)\Delta x$. In practice, of course, one would want to use much smaller increments for greater accuracy. For a range of x_1 from 9 to 17, we calculate the f(x)dx as $P(x_1-.25 < X < x_1+.25)$ using a normal distribution with mean 13 and standard deviation 1 (as in the simulation) in Column 2 and label them P[x_1+-.25]. Column 3 contains the probabilities P(D| x_1) calculated previously (see Column 4 of previous table). The product of Column 2 and Column 3 is in Column 4. The Column 4 values are used in the numerator of (3), while the Column 2 values are used in the denominator.

Table 2: Sensitivity of P(D) to Changes to X_1's Distribution

	N(13,1)			x_1 Limits			N(14,.5)	Gamma
x_1	P[x_1+-.25]	P(D|x_1)	P[x_1+-.25]*P(D|x_1)	9-17	11-15	12-14	P[x_1+-.25]	P[x_1+-.25]
9	0.0000777	0.0137	0.0000011	1	0	0	0.0000000	0.0000000
9.5	0.0004886	0.0104	0.0000051	1	0	0	0.0000000	0.0000000
10	0.0024027	0.0077	0.0000185	1	0	0	0.0000000	0.0000000
10.5	0.0092447	0.0062	0.0000573	1	0	0	0.0000000	0.0000003
11	0.0278347	0.0045	0.0001253	1	1	0	0.0000000	0.0000156
11.5	0.0655906	0.0037	0.0002427	1	1	0	0.0000034	0.0003835
12	0.1209776	0.0025	0.0003024	1	1	1	0.0002292	0.0046993
12.5	0.1746663	0.0021	0.0003668	1	1	1	0.0059770	0.0303462
13	0.1974127	0.0016	0.0003159	1	1	1	0.0605975	0.1084127
13.5	0.1746663	0.001	0.0001747	1	1	1	0.2417303	0.2235748
14	0.1209776	0.0009	0.0001089	1	1	1	0.3829249	0.2762957
14.5	0.0655906	0.0008	0.0000525	1	1	0	0.2417303	0.2115208
15	0.0278347	0.0011	0.0000306	1	1	0	0.0605975	0.1033464
15.5	0.0092447	0.0012	0.0000111	1	0	0	0.0059770	0.0331060
16	0.0024027	0.0015	0.0000036	1	0	0	0.0002292	0.0071263
16.5	0.0004886	0.0021	0.0000010	1	0	0	0.0000034	0.0010543
17	0.0000777	0.003	0.0000002	1	0	0	0.0000000	0.0001095

We now use the table to calculate the probability of a defective Y for truncation based on three sets of tolerance limits on x_1. We calculate P(D) when x_1 is allowed to range from 9 to 17, from 11 to 15, and from 12 to 14 using the indicator functions given in Columns 5, 6, and 7 respectively. Multiplying Col-

umns 2 (for the denominator) and 4 (for the numerator) with the appropriate indicator function, one can calculate P(D) for each set of tolerance limits. For example, we calculate the probability P(D) when x_1 is allowed to range from 12 to 14 as follows:

$$P(D) = \frac{0.0003024 + 0.0003668 + 0.0003159 + 0.0001747 + 0.0001089}{0.1209776 + 0.1746663 + 0.1974127 + 0.1746663 + 0.1209776} = .00161.$$

Thus, 0.00161 is the estimate of the probability of a defective if only values of x_1 in the range from 12 to 14 are used. For comparison, similar values have been calculated for truncation to the tolerance limits of 11 to 15 and for the tolerance limits of 9 to 17 and are shown below:

X_{1L}	X_{1U}	P(D)
9	17	.00182
11	15	.00176
12	14	.00161

Note that only one simulation is necessary to evaluate any number of tolerance limits for any X_i. Although the calculations assume that the other input variables are not altered, one could perform additional simulations with updated tolerance limits to achieve a desired capability in iterative simulation steps.

We emphasize that, once the simulation has estimated $P(D \mid X_i = x_i)$ at the selected values for x_i, (3) could be evaluated using **any** probability distribution for X_i. Hence, without conducting any more simulation replications, we can evaluate P(D) for any probability distribution of any X_i. To be precise, we mean we can do this for any X_i as long as we don't change the probability distributions for any other X_j, $j \neq i$. This is because $P(D|X_i = x_i)$ depends on the probability distributions for all other X_j, $j \neq i$ but is **independent** of the probability distribution of X_i. In terms of this example, the values for column 3 are independent of the probability distribution for X_1. One can thus estimate P(D) for any distributional model /parameters / truncation simply by updating the column 2 values to reflect that distribution. For example, in column 8 we show the discretized probabilities for the normal distribution with a mean of 14 and a standard deviation of .5. If you replace column 2 with these values and compute P(D) with the distribution truncated from 13 to 15 (mean of 14 +/- 1), P(D)=.00096. In this example, the effects of simultaneous shift (mean from 13 to 14), narrow (standard deviation decreased from 1 to .5), and chop (truncation at mean +/- 1) actions are evaluated with no additional simulation replications required. Column 9 shows the discretized probabilities for the gamma distribution with parameters chosen so that the mean is 14 and the standard deviation is .5. Replacing column 2 with these values and truncating from 13 to 15, yields P(D)=.00101. This example shows the ultimate flexibility of the approach; P(D) is evaluated for a completely different distributional form accompanied by shift, narrow, and chop actions with no additional simulations required.

The third ability we mentioned is the ability to efficiently produce a graph of the probability that Y is defective as a function of any parameter of the probability distribution of Y. This can be done by changing the parameter value in increments across a desired range, updating the discretized probabilities for Y to reflect each value, and using the approach just described to evaluate P(D) for each value of the parameter in the range. The function can then be plotted using the calculated values. Although this requires the computations to be done as many times as there are parameter values in the range, it is important to note that no additional simulation runs need to be conducted.

7   CONCLUSIONS AND RECOMMENDATIONS

We have presented an approach that allows the effects of significant (rather than infinitesimal) changes to input variable distributions on the probability that output variables will be outside specification limits to be efficiently assessed. Changes include shifting the mean, narrowing the distribution (lowering the

standard deviation), chopping or truncating the distribution, any combination of the three, and even complete changes to the distributional form. We have demonstrated the approach using simple numerical examples. The ability to handle correlated input variables and/or simultaneous changes to multiple input variables would be useful extensions.

REFERENCES

Boning, D.S., and P. K. Mozumder. 1994. "DOE/Opt: a System for Design of Experiments, Response Surface Modeling, and Optimization Using Process and Device Simulation." IEEE Transactions on Semiconductor Manufacturing 7:233-244.

Bowman, A., and J. Schmee. 2004. "Estimating Sensitivity of Process Capability Modeled by a Transfer Function." Journal of Quality Technology 36:223–239.

Bowman, A., and J. Schmee. 2011. "Extensions of Shift, Narrow, and Chop to Multiple Outputs with Non-monotonic Transfer Functions." Working Paper, School of Management, Union Graduate College, Schenectady, NY.

Bowman, A., N. Doganaksoy, and J. Schmee. 2010. "Efficient Gradient-Based Optimization of Process Capability with Multiple Potentially Nonnormal Outputs." *Communications in Statistics – Simulation and Computation* 39:807-827.

Carlyle, W. M., D. C. Montgomery, and G. C. Runger. 2000. "Optimization Problems and Methods in Quality Control Improvement (with Discussions)." *Journal of Quality Technology* 32: 1–31.

Chiao, C., and M. Hamada. 2001. "Analyzing Experiments with Correlated Responses." *Journal of Quality Technology* 33: 451–465.

Creveling, C. M. 1997. *Tolerance Design: A Handbook for Developing Optimal Specifications*, Reading, MA: Addison Wesley Longman Inc.

Evans, D. H. 1975. "Statistical Tolerancing: the State of the Art. Part III: Shifts and Drifts." *Journal of Quality Technology* 7:72-76.

JMP. 2009. *JMP 8 Statistics and Graphics Guide*. Chapter on Profiling, Section on The Simulator. Cary, NC: JMP–A Business Unit of SAS.

Kalil, S. J., F. Maugeri, and M. I. Rodrigues. 2000. "Response Surface Analysis and Simulation as a Tool for Bioprocess Design and Optimization." *Process Biochemistry* 35-6:539-550.

Kotz, S., and N. L. Johnson. 1993. *Process Capability Indices*. London: Chapman and Hall.

Kumar, U. D. , D. Nowicki, J. E. Ramírez-Márquezb, and D. Vermab. 2008. "On the Optimal Selection of Process Alternatives in a Six Sigma Implementation." *International Journal of Production Economics* 111:456-467.

Law, A. M., and W. D. Kelton, 1991. *Simulation Modeling and Analysis*. New York, NY: McGraw-Hill.

Myers, R. H., A. I. Khuri, and W. H. Carter, Jr. 1989. "Response Surface Methodology: 1966-1988." *Technometrics* 31:137 – 157.

Myers, R. H., and D. C. Montgomery. 1995. *Response Surface Methodology*. New York, NY: Wiley.

Myers, R. H., D. C. Montgomery, G. G. Vining, C. M. Borror, and S. M. Kowalski. 2004. "Response Surface Methodology: A Retrospective and Literature Survey. *Journal of Quality Technology* 36:53 –77.

Myers, R. H., D. C. Montgomery, and C. M. Anderson-Cook. 2009. *Response Surface Methodology: Process and Product Optimization Using Designed Experiments*. NY: Wiley.

Polansky, A. M. 2001. "A Smooth Nonparametric Approach to Mulivariate Process Capability." *Technometrics* 43(2):199-211.

Skowronski, V. J., and J. U. Turner. 1997. "Using Monte-Carlo Variance Reduction in Statistical Tolerance Synthesis." *Computer-Aided Design* 29:63-69.

Snee, R. D., and R. W. Hoerl. 2003. *Leading Six Sigma: A Step by Step Guide*. Upper Saddle River, NJ: FT Prentice Hall.

Spring, F., B. Leung, S. Cheng, and A. Yeung. 2003. "A Bibliography of Process Capability Papers." *Quality and Reliability Engineering International* 19(5):445–460.

Wu, C. F. J., and J. Hamada. 2000. *Experiments: Planning, Analysis, and Parameter Design Optimization*. New York, NY: Wiley.

Wu, C.-W., W. L. Pearn, and S. Kotz. 2009. "An Overview of Theory and Practice on Process Capability Indices for Quality Assurance." *International Journal of Production Economics* 117(2):338-359.

AUTHOR BIOGRAPHIES

ALAN BOWMAN is Professor of Management at Union Graduate College in Schenectady, NY. He received his Ph.D. in Operations Research from Cornell University in 1990. His research interests focus on the use of probability models, statistical models, and simulation in the areas of quality management, inventory management, project management, and production scheduling. He is an active management consultant helping companies use models to improve quality and productivity. He has published in many journals including *Management Science, Operations Research, Production and Operations Management, Naval Research Logistics, and Journal of Quality Technology*. His email address is bowmana@uniongraduatecollege.edu.

JOSEF SCHMEE is retired Professor of Management from Union College in Schenectady, NY. He received his Ph.D. in Engineering and Administrative Systems from Union College in 1974. His research interests focus on six sigma quality improvement, life data analysis, design of experiments and statistical process control. He is also an active consultant helping companies in these areas He has published in many journals including *Journal of the American Statistical Association, Technometrics, Communications in Statistics, and Journal of Quality Technology*. He is co-author (along with Jane Oppenlander) of the book JMP Means Business: Statistical Models for Management. His email address is jschmee@gmail.com.