

THE IMPORTANCE OF MODELING AND SIMULATION IN DFSS

*Jesse Peplinski, Vice President
Statistical Design Institute, LLC*

Abstract – The fundamental objective of Design for Six Sigma (DFSS) is to design products and processes that meet your customers’ needs, cost effectively, without production or integration problems. This objective can be achieved by following a simple set of good design practices: understand your customers’ needs, select the design concept that is most likely to succeed, mathematically predict the cost and performance of your design, and then make design improvements before committing to capital purchases and supplier contracts. There can of course be pitfalls in applying each of these practices, but the most common weaknesses we have seen in DFSS implementations are in the area of mathematical prediction. In this paper I will show how modeling and simulation can be used to overcome these weaknesses and reap the full benefits of DFSS.

1 Foundation: Configuring DFSS for Success

Implementations of DFSS can take many forms. While all share the common thread of the good design practices above, most implementations lock them into a sequence of phases like DMADV (Design, Measure, Analyze, Design, Validate) or IDOV (Identify, Design, Optimize, Validate). Company-specific acronyms are also common. More often than not, these sequences exist separately from the company’s established development processes, so extra overhead of “aligning” the DFSS acronym with the development process often becomes necessary. A solution to this complexity is to recognize that any development process is a series of tasks, and that the DFSS good design practices fit within every task. Implementing DFSS then becomes completely independent from the specific development process, and instead can be deployed as an enabler to the process itself. This type of implementation is shown in Figure 1.

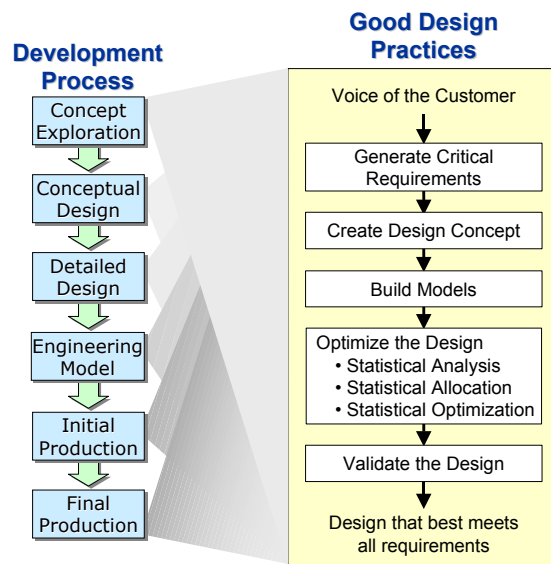


Figure 1: Integrating DFSS into a Development Process

The design tasks of a typical development process are shown down the left side of Figure 1, with the tasks grouped under the headings of “Concept Exploration”, “Conceptual Design”, and so on. These specific headings are somewhat arbitrary and will change from one implementation to another, but the activities within them are largely universal. In Concept Exploration and Conceptual Design many different design concepts are generated and evaluated. Each “concept” is typically a high-level layout of the different functions or subsystems of the design, with selected components or technologies for each. These concepts exist primarily on paper (or on computer). In Detail Design and Engineering Model the selected best concept(s) are fleshed out in more detail, again primarily on paper. Physical prototypes or mock-ups will be built to test and verify the calculations behind key features of the design, but the expense of prototypes is kept as small as possible. Finally in Initial Production and Final Production the completed design is implemented and released to your customers, perhaps in large quantities.

A design task typically culminates in a design decision – the selection of a technology or component or material, the determination of the best dimensions or parameter values, and so on. One by one, these decisions push the design forward and flesh out its details. Ideally these decisions are made in the best interests of your customers and stakeholders, both internal and external. Following the good design practices on the right side of Figure 1 helps accomplish this goal. The “Voice of the Customer” is data collected directly from your customers through interviews, focus groups, surveys, warranty and complaint data, enhancement requests, etc. From this unfiltered data the true customer needs are determined, and these are used to identify the “Critical Requirements” for the design – quantifiable targets and characteristics that can be tested and measured on the design itself. The requirements should be stated generically enough to encourage the creation of several different potential solutions for each element of the design. The “Design Concept” is then the combination of all of the selected best solutions.

Once the design concept is assembled a determination can then be made as to whether it is affected by variation. If it will be made in large quantities, will all units perform the same the same way every time? Even for just one unit, will it be used in precisely the same manner and in the same environmental conditions? Will its performance remain constant over the life of the design? Typically the answer to at least one of these questions is “no”. Physical components and materials vary from one unit to the next and degrade over time. Process parameters shift and drift over time, and environmental conditions can seldom be controlled very precisely.

If variation must be considered, it is not sufficient to predict the design’s performance deterministically. In other words, having one prototype that works under controlled conditions does not prove that the design will perform well under other conditions or over time. Instead a statistical analysis is used to assess the performance of the design across the complete range of variation. From this analysis an estimate of the *probability* of the design performing acceptably can be determined. There are two ways in which this analysis can be performed: build many samples and test and measure their performance, or predict the design’s performance mathematically. For the obvious reasons of time and expense, mathematical prediction is often the only viable option. Therefore it becomes a crucial step in DFSS to create mathematical models of the design.

2 Types of Mathematical Models for DFSS

A mathematical model of a design can take many forms. It can be an equation from a textbook (physics, engineering, finance, accounting, etc.), a computer simulation, a prototype whose performance can be measured, or a set of historical data. Regardless of the type of model it can be represented as a “black box” as is shown in Figure 2.

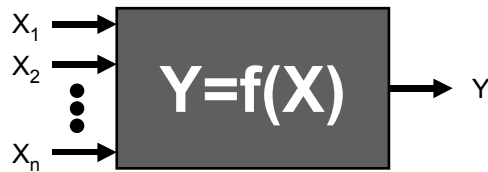


Figure 2: A “Black Box” Representation of a Model

The output of the black box, “Y”, is a quantifiable parameter that maps directly to a critical requirement from your customer. Examples of such output parameters are weight, efficiency, cycle time, cost, and so on. (A model may have more than one “Y”, but for simplicity only one is shown here.) The inputs to the black box are parameters “ X_i ” that characterize the given design. Examples of these input parameters are material properties, process settings, dimensions, component values, etc. When the inputs are specified one can “turn the crank” and generate a value for the output Y. This output parameter is deterministic and is completely determined by the model’s input parameter values. Notice that there is no distinction between “control factors” or “noise factors” here; all are classified as X’s. The mathematics of the black box is shown as “ $Y=f(X)$ ”, but the math may not actually be an explicit equation. For example, different prototypes can be built with different parameter values X_i , and their outputs Y can be measured instead of calculated.

Statistical analyses such as Sensitivity Analysis or Monte Carlo analysis are performed on the black box by changing the input parameter values and observing the changes on the output parameter. When a sufficient number of output values have been collected, a probability distribution can be constructed for the output parameter, and this distribution will tell us the likelihood of the design satisfying the customer requirement. More details of these analyses will be given in the next section, but for now it is enough to note that typically large numbers of output values (on the order of hundreds or thousands) will be required. It therefore becomes important that the output values can be generated quickly and cost-effectively.

However, not all models can be computed in a cost-efficient manner. Prototype designs can be very expensive to produce and measure. Finite-element simulations of a design can take hours or days to generate a single output Y value. In these cases statistical modeling techniques can be applied to create fast, accurate approximations of the original model. A decision tree to guide users through this statistical modeling process is shown in Figure 3. All of the different categories of models are listed down the left side of the figure – equations, data, simulations, prototypes, and the actual system. These represent the actual models gathered or generated to predict each of the design’s critical requirements. Observe that for all cases it is possible to generate fast, accurate approximations if necessary.

In Figure 3 there are three paths shown. For the first path, existing equations (from textbooks, from expert judgement, etc.) typically will compute very quickly with today’s computer tools.

As long as their accuracy is sufficient, they can be used for DFSS as is. (If their accuracy is not sufficient then that may initiate a separate activity of improving them or perhaps choosing a different type of model to predict the critical requirement.) The second path is for models that exist as data sets. These data sets typically are a set of measurements taken over time or over different production units where, for each unit, all of the input X's and the output Y's are measured. The standard format for this data is shown on the left of Figure 4. By applying regression analysis a fitted equation can be generated from the data, and a simple equation is shown on the right of Figure 4.

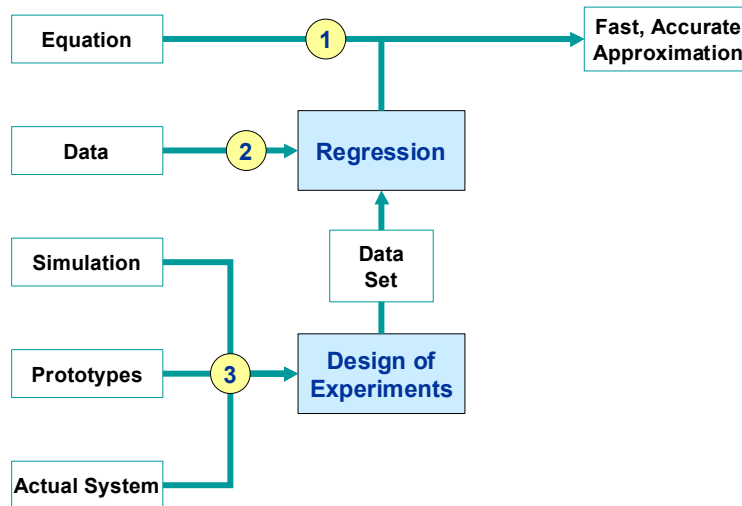


Figure 3: Decision Tree for Statistical Modeling

Computer simulations, prototypes and the actual system do not have explicit mathematics, but if they could generate a set of data then regression analysis could be applied. As is shown by the third path of Figure 3, the Design of Experiments is a powerful way to efficiently collect a small set of data from these expensive sources, and from this data a fitted equation can be built for use in DFSS analyses. In this section we have discussed the types of mathematical models, the importance of having models that compute quickly, and the techniques for creating fast approximations for every model type. In the next section we will show the benefits of such fast models in terms of statistical analysis, statistical allocation, and multi-objective optimization.

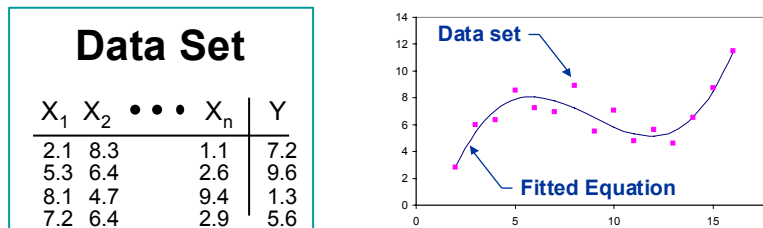


Figure 4: Transforming a Data Set into a Fitted Equation

3 Benefits of Modeling and Simulation

If a model can be created to predict your design's performance with respect to a critical requirement, and if this model can be computed relatively quickly, then powerful statistical analyses become available that allow you to reap the full benefits of DFSS. You can predict the

probability of the design meeting the requirement given environmental variation, manufacturing variation, and usage variation. If this probability is not sufficiently large then you can determine the maximum allowable variation on the models inputs to achieve the desired output probability. And if the input variation can not be controlled, you can explore new input parameter values that may improve your design’s statistical performance with respect to multiple requirements simultaneously. We will call these techniques “Statistical Analysis”, “Statistical Allocation”, and “Statistical Optimization”, respectively.

3.1 Statistical Analysis

With a typical model a single value is specified for each input parameter, and a single value is computed for each output parameter. However, in real-world applications each input parameter will vary over time or from one unit to the next. It is imperative while designing to capture all of these sources’ variation and to ensure that the output parameter behaves as needed across all combinations of input parameter values. This problem is represented in Figure 5 as a black-box model with a probability distribution specified for each input parameter. As these probability distributions filter through the mathematics of the model, they generate a probability distribution for each output parameter. If the mathematics is nonlinear, then the shape of the output distribution may not bear a resemblance to any of the input distributions.

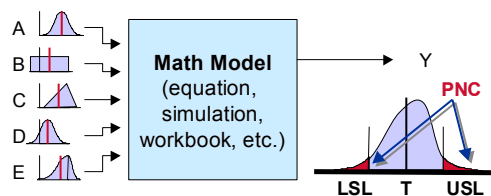


Figure 5: Black-Box Representation of Statistical Analysis

When the output parameter maps to a customer requirement, typically there are numerical limits defined that represent acceptable or unacceptable values. For an output parameter like weight or cost or cycle time, often there will be an upper limit that when exceeded would lead to customer dissatisfaction. For an output parameter like efficiency or speed or reliability, often there will be a lower limit that if not achieved would lead to customer dissatisfaction. There are also parameters like resonant frequencies or delivery times that may have both upper and lower limits. These limits are shown as the Upper Specification Limit (USL) and Lower Specification Limit (LSL) in Figure 5. The areas of the output probability distribution that fall outside these limits represent the probability of non-compliance (PNC) of the requirement. PNC can be a very useful metric in DFSS for tracking and improving the design’s performance in terms of customer satisfaction.

The central challenge in this type of statistical analysis is generating the probability distribution for the output parameter. Many different techniques exist, but two of the most prevalent are Sensitivity Analysis and Monte Carlo Analysis. Sensitivity Analysis approximates the output distribution by taking a low-order Taylor’s Series expansion of the mathematical model and combining it with the low-order moments of the input distributions to compute the low-order moments of the output distribution. From these moments the location, spread and sometimes the shape of the output distribution can be determined. Monte Carlo Analysis approximates the output distribution by randomly generating single values for each of the input parameters,

plugging these values into the model, and computing a value for the output parameter. This process is repeated hundreds or thousands of times, generating a large sample of output values. By then applying sample statistics a substantial amount of information can be derived about the output distribution – its location, spread and shape. Monte Carlo Analysis is the gold standard to which all other techniques are judged, and if computational expense is not a problem, then it will always be the preferred method. Sensitivity Analysis often is just as accurate as Monte Carlo Analysis, and it typically requires fewer model calculations to generate results. If the model is slow or expensive to compute, then Sensitivity Analysis is often used.

3.2 Statistical Allocation

Once a statistical analysis is performed, often the PNC value that results is unacceptably high. In these cases, one way to improve the PNC value is to apply statistical allocation. Allocation techniques use Sensitivity Analysis or Monte Carlo analysis to work the problem backward and identify the input factor standard deviations (roughly a measure of the spread of their distributions) that will reduce the PNC to a desired level. An illustration of statistical allocation is shown in Figure 6. An infinite number of closed-form solutions (vectors of input standard deviations) can be generated if the allocation is based on Sensitivity Analysis techniques. If Monte Carlo analysis is employed, then allocation becomes a brute-force iterative process.

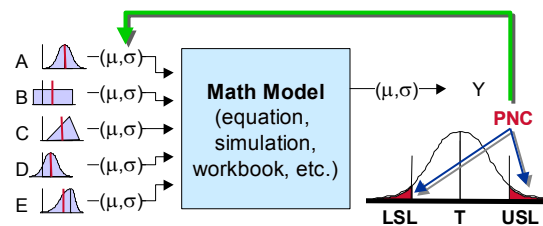


Figure 6: Black-Box Representation of Statistical Allocation

3.3 Statistical Optimization

Improving an output parameter’s PNC by reducing its input parameters’ variation is seldom ideal. Often it is preferable to reduce an output’s PNC while keeping the input variation constant, thus making the output *robust* to input parameter variation. This can be accomplished by searching for new mean values for each input parameter – in essence shifting the distributions higher or lower but keeping the width of the distributions the same. This can easily be done for multiple output parameters simultaneously by employing multi-objective statistical optimization techniques. A graphical representation of multi-objective, statistical optimization is shown in Figure 7.

To apply optimization techniques, the first step is to build a formulation of the problem that captures all of the input parameters and output parameters. Each input parameter’s mean value can be specified as fixed or as searchable, and in both cases variation can be specified. Searchable input parameters can be defined three ways: as continuous, integer, or discrete. A continuous parameter can take on any values between a specified minimum and a specified maximum. Integer parameters can take on only integer values between a minimum and a maximum, and discrete parameters can take on only values specified in a user-defined list.

Optimization software tools automatically try different values for the input parameters in an attempt to improve the design's output parameters. "Improvement" is defined by constraints and goals in the formulation. Constraints typically take an output's mean, standard deviation or PNC and specify a threshold value that it must achieve. Goals typically take a similar value and define a target value that it will ideally achieve. By definition constraints are a higher priority than goals. The logic behind the search sequence of input parameter mean values is contained in the software tool's algorithm, typically either gradient-based or heuristic. The output of a statistical optimization problem is a vector of input parameters mean values and/or standard deviations that bring the critical requirements' means, standard deviations, and PNC values as close as possible to their customer-specified targets.

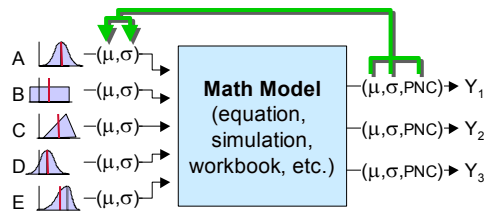


Figure 7: Black-Box Representation of Statistical Optimization

4 Statistical Optimization Example

The typical examples we see in DFSS applications contain several nonlinear models that map into multiple, conflicting objectives across the dimensions of design performance, cost and reliability. Therefore it is our standard recommendation to employ optimization packages and algorithms that can perform a global, nonlinear, multi-objective, statistical search. One such example that works well for engineering audiences is the design of a low-pass filter. This is a component selection problem where the circuit configuration is given and the mathematical models are well-understood. A schematic of the filter to be designed is shown in Figure 8.

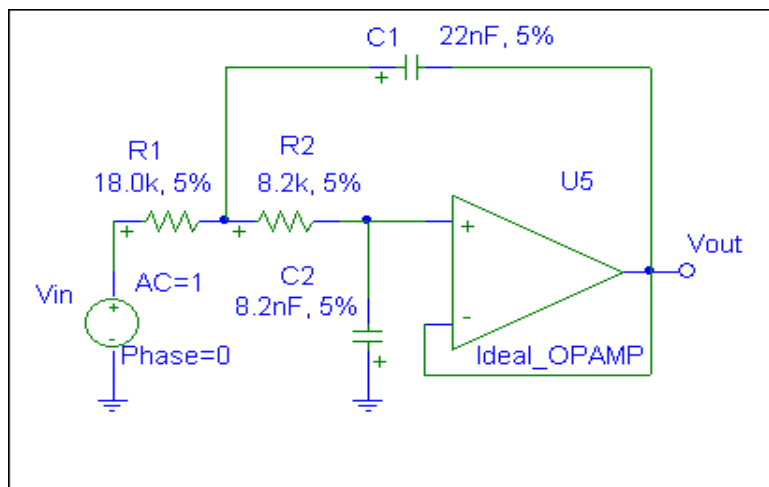


Figure 8: Schematic of a Low-Pass Filter

In this design problem resistors must be chosen for R1 and R2, capacitor values must be chosen for C1 and C2, and an op amp must also be selected. For each resistor and capacitor a nominal value must be specified along with a "percent tolerance". When components are purchased in

bulk, the supplier will warranty that all individual values fall within plus-or-minus some percentage of the nominal value. If we assume that all values are uniformly distributed, then an equation can be derived for the component's standard deviation. In addition, we also have the ability to select components with different published failure rates. Components with higher failure rates are cheaper but less reliable. Because we wish to purchase standard components from published catalogs, we do not have the luxury of specifying custom values for each component. Instead we must select our components from standard published lists. A summary of all of this design parameter information is shown in Table 1.

X	Name	Type	Standard Deviation	Feasible Mean Values	Units
1	R1	Discrete	$= \frac{R\%Tol \cdot R1}{100\sqrt{3}}$	6800; 7500; 8200; 9100; 10000; 11000; 12000; 13000; 15000; 16000; 18000; 20000; 22000; 24000	Ohms
2	R2	Discrete	$= \frac{R\%Tol \cdot R2}{100\sqrt{3}}$	6800; 7500; 8200; 9100; 10000; 11000; 12000; 13000; 15000; 16000; 18000; 20000; 22000; 24000	Ohms
3	C1	Discrete	$= \frac{C\%Tol \cdot C1}{100\sqrt{3}}$	1e-10; 1.2e-10; 1.5e-10; 1.8e-10; 2.2e-10; 2.7e-10; 3.3e-10; 3.9e-10; 4.7e-10; 5.6e-10; 6.8e-10; 8.2e-10; 1e-9; 1.2e-9; 1.5e-9; 1.8e-9; 2.2e-9; 2.7e-9; 3.3e-9; 3.9e-9; 4.7e-9; 5.6e-9; 6.8e-9; 8.2e-9; 1e-8; 1.2e-8; 1.5e-8; 1.8e-8; 2.2e-8; 2.7e-8; 3.3e-8; 3.9e-8; 4.7e-8; 5.6e-8; 6.8e-8; 8.2e-8	Farads
4	C2	Discrete	$= \frac{C\%Tol \cdot C2}{100\sqrt{3}}$	1e-10; 1.2e-10; 1.5e-10; 1.8e-10; 2.2e-10; 2.7e-10; 3.3e-10; 3.9e-10; 4.7e-10; 5.6e-10; 6.8e-10; 8.2e-10; 1e-9; 1.2e-9; 1.5e-9; 1.8e-9; 2.2e-9; 2.7e-9; 3.3e-9; 3.9e-9; 4.7e-9; 5.6e-9; 6.8e-9; 8.2e-9; 1e-8; 1.2e-8; 1.5e-8; 1.8e-8; 2.2e-8; 2.7e-8; 3.3e-8; 3.9e-8; 4.7e-8; 5.6e-8; 6.8e-8; 8.2e-8	Farads
5	%Tolerance for resistors R%T	Discrete	0	1; 2; 5	%
6	%Tolerance for capacitors C%T	Discrete	0	1; 2; 5	%
7	R failure rate λ_R	Discrete	0	.0022; .0037; .051; .07	failures per million hours
8	C failure rate λ_C	Discrete	0	0.001; 0.0017; 0.039; 0.064	failures per million hours
9	Op amp failure rate λ_{OP}	Discrete	0	0.033; 0.11	failures per million hours

Table 1: Design Parameter Information

The target performance requirements for the circuit are that it must maintain a cutoff frequency of 1000 Hertz, and the slope of its frequency response shape should follow the ideal Butterworth value of 0.707. Customer satisfaction should be maintained if these targets can be met within $\pm 10\%$. In addition, the customer would like the reliability of the circuit expressed in terms of its 10-year life, and lower costs are preferred. Mathematical models for the design's performance can be found in engineering textbooks, and a model for the probability of the design lasting 10 years without failure can be derived by applying an exponential failure rate to each component and by summing up the components in series. Component costs can fluctuate, so instead of trying to predict costs in absolute units a relative cost index is derived, where lower costs are given to components with higher percent tolerances and higher failure rates. A summary of the mathematical models employed is shown in Table 2.

Y	Name	Lower Spec Limit (LSL)	Formula Y = f(X)	Upper Spec Limit (USL)
----------	-------------	-------------------------------	-------------------------	-------------------------------

1	Cutoff frequency F_c	900	$F_c = \frac{1}{2\pi} \cdot \frac{1}{\sqrt{R_1 \cdot C_1 \cdot R_2 \cdot C_2}}$	1100
2	Frequency Response shape Q	0.636	$Q = \frac{1}{\sqrt{\frac{R_1 \cdot C_2}{R_2 \cdot C_1} + \frac{R_2 \cdot C_2}{R_1 \cdot C_1}}}$	0.778
3	Cost (relative cost index)		$Cost = \frac{1}{R\%T} + \frac{1}{C\%T} + \frac{1}{1000\lambda_R} + \frac{1}{100\lambda_C} + \frac{1}{10\lambda_{OP}}$	
4	Reliability as P(Life > 10 years)		$P(\text{life} > 10 \text{ years}) = e^{-\left(\frac{10 \cdot 24 \cdot 365 \cdot (2\lambda_R + 2\lambda_C + \lambda_{OP})}{1,000,000}\right)}$	

Table 2: Mathematical Models and Specifications

The next step in building the optimization problem is to map the multiple conflicting requirements into constraints and goals. We have found that the best way to handle multi-objective problems like this is to construct different *scenarios* that vary the relative priorities of the objectives. In this fashion a Pareto-optimal *family* of design solutions can be generated, trading off cost with performance and reliability. Following this logic there is no one “optimal” solution, and instead the design team as a whole will choose the “balanced” design that best meets all relative customer preferences. A list of four different scenarios is shown in Table 3.

Scenario	Balanced Cost	Balanced Reliability	High Performance	High Reliability
Constraints	$F_c \text{ PNC} \leq 0.001$ $Q \text{ PNC} \leq 0.001$	$F_c \text{ PNC} \leq 0.001$ $Q \text{ PNC} \leq 0.001$		$F_c \text{ PNC} \leq 0.001$ $Q \text{ PNC} \leq 0.001$
Priority 1 Goals	Cost Mean = 0	P(Life) Mean ≥ 0.98	$F_c \text{ PNC} \leq 3.4\text{E-}6$ $Q \text{ PNC} \leq 3.4\text{E-}6$	P(Life) Mean ≥ 0.999
Priority 2 Goals	P(Life) Mean ≥ 0.98	Cost Mean = 0	Cost Mean = 0	Cost Mean = 0
Priority 3 Goals	$F_c \text{ PNC} \leq 3.4\text{E-}6$ $Q \text{ PNC} \leq 3.4\text{E-}6$	$F_c \text{ PNC} \leq 3.4\text{E-}6$ $Q \text{ PNC} \leq 3.4\text{E-}6$	P(Life) Mean ≥ 0.98	$F_c \text{ PNC} \leq 3.4\text{E-}6$ $Q \text{ PNC} \leq 3.4\text{E-}6$

Table 3: Multi-Objective Scenarios

To perform the optimization search we employ Apogee, an Excel-based software tool that is part of the SDI Tools suite. With Apogee all of the design parameters, mathematical models, constraints and goals are easily specified in Excel workbooks. Running different scenarios is accomplished simply by changing the goal priorities and targets as desired. A screenshot of the “Balanced Cost” scenario’s input is shown in Figure 9, and a summary of the results from all of the scenarios is shown in Table 4.

Notice in Table 4 that there are multiple combinations of resistor and capacitor values that yield nominal cutoff frequencies near 1000 Hertz and nominal frequency response shapes near 0.707. In the “Balanced Cost” solution it is possible to use the lowest-cost components and still achieve acceptable performance (PNC’s of 0.001 or less) and a 96% chance of a 10-year life. If higher performance is desired, then the circuit’s PNC’s can be reduced to better than six-sigma levels of quality by using 2% components at a higher cost. Finally if reliability is pursued above all else then the probability of a 10-year life can be increased to 99.6% with an even greater impact on component costs. *Which solution is best?* That judgment is best made by the design team, weighing the relative trade-offs between performance, cost and reliability.

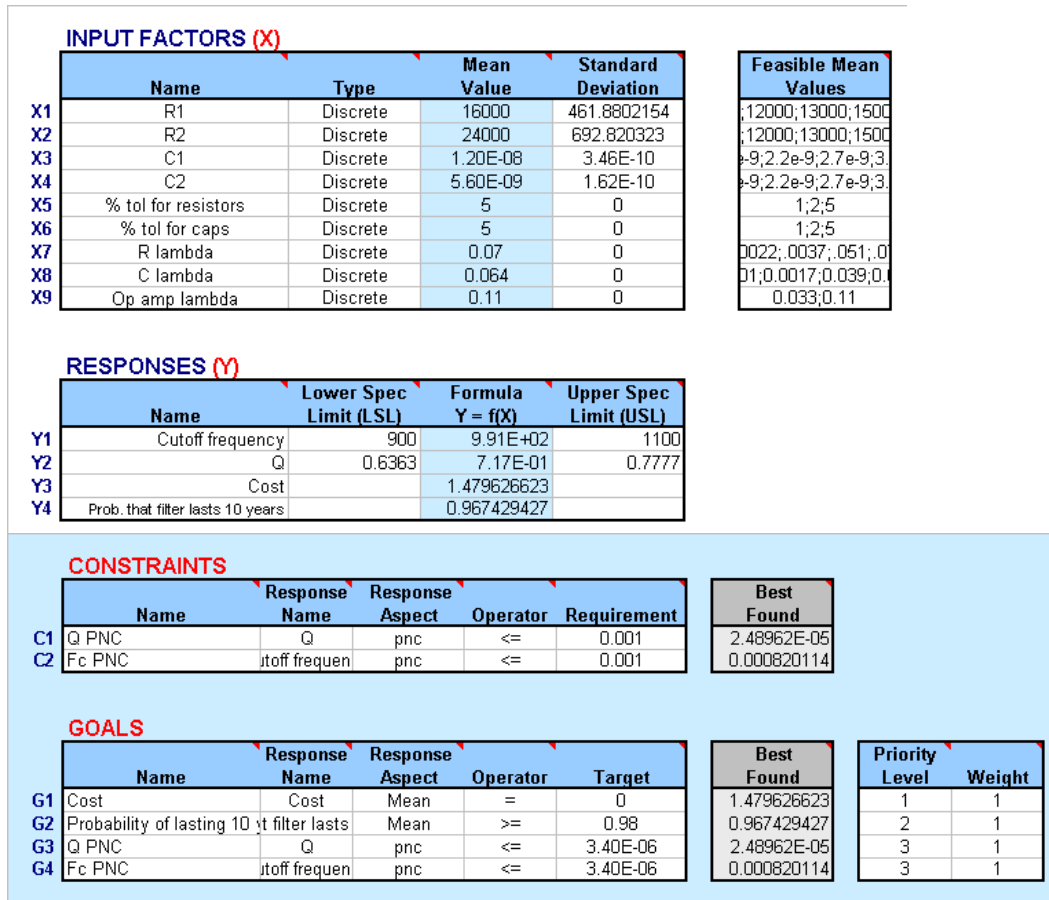


Figure 9: Apogee Screenshots for Scenario 1 Inputs

Solution	Balanced Cost	Balanced Reliability	High Performance	High Reliability
R1	16000	16000	16000	16000
R2	24000	13000	24000	24000
C1	1.20E-08	1.50E-08	1.20E-08	1.20E-08
C2	5.60E-09	8.20E-09	5.60E-09	5.60E-09
R%T	5	5	2	5
C%T	5	2	2	5
λ_R	0.07	0.0037	0.07	0.0022
λ_C	0.064	0.039	0.064	0.001
λ_{OP}	0.11	0.11	0.11	0.033
Results				
F _C Mean	990.7632359	995.0293896	990.7632359	990.7632359
F _C Std Dev	28.60087123	21.87556531	11.44034849	28.60087123
F _C PNC	0.000820114	7.79192E-06	1.06435E-15	0.000820114
Q Mean	0.717137166	0.672624021	0.717137166	0.717137166
Q Std Dev	0.014928401	0.005672643	0.00597136	0.014928401
Q PNC	2.48962E-05	7.59933E-11	1.7943E-24	2.48962E-05
Cost Mean	1.479626623	2.135771436	2.079626623	13.88484848
P(Life) Mean	0.967429427	0.983028624	0.967429427	0.996554509

Table 4: Multi-Objective Solutions

What can we conclude from this example? First, there is tremendous power in building mathematical models to predict a design's behavior. Even simple textbook equations display nonlinear wrinkles. Second, in the early stages of a design's development it is not difficult to build relative models for cost and reliability. These models can be very effective in driving the optimization towards low-cost or high-reliability solutions. Third, in a multi-objective environment it is important to generate a family of Pareto-optimal solutions for the design team to review. Do NOT give the software the ultimate authority and assume that there is just one "optimal" solution. And finally, although the low-pass filter is relatively simple, we have had similar success applying modeling, simulation and optimization to larger, system-level problems across many design disciplines. We encourage you to do the same.

5 Summary

The fundamental objective of Design for Six Sigma (DFSS) is to design products and processes that meet your customers' needs, cost effectively, without production or integration problems. In this paper we have discussed how this objective can be achieved by configuring DFSS as a simple set of good design practices. A key step in the process is to use modeling and simulation to create "black box" predictive models for your design's critical requirements, and statistical modeling techniques such as regression and the design of experiments can help ensure that the mathematics are explicit and compute quickly. With these models the design can be statistically analyzed and optimized to ensure it will perform as expected while being robust to the variation inherent in its production, environment and use. In this manner the full benefits of DFSS can be realized.

6 About the Author

Jesse Peplinski is Vice-President of Technology and Development for the Statistical Design Institute, LLC and is a certified Master Black Belt. SDI is a leading provider of training programs and software tools for implementing DFSS in product development, and Jesse is one of SDI's primary training instructors. He also leads the development of their DFSS software tools and training materials. Jesse has worked as a Master Black Belt at General Electric Medical Systems, guiding the CT engineering organization in the daily application and implementation of DFSS and six sigma. As a systems engineer at Texas Instruments and Raytheon Systems Company he helped establish a foundation for applying Six Sigma concepts during requirements development and concept modeling, well before hardware is built. Jesse received his Ph.D. and M.S. in Mechanical Engineering from Georgia Tech in 1997 and 1994, respectively, and he received his B.S. in Engineering Science from Harvard University in 1992. You can contact him at jpeplinski@stat-design.com or visit us at www.stat-design.com.

