# SIMULATION AND OPTIMIZATION AS EFFECTIVE DFSS TOOLS

Karl Luce
Lucie Trepanier
Fred Ciochetto
Lawrence Goldman

1515 Arapahoe Street, Suite 1311
Decisioneering, Inc.
Denver, CO 80202, U.S.A.

## ABSTRACT

Simulation and optimization techniques can provide Design for Six Sigma (DFSS) practitioners with reduced reliance on physical prototypes, rapid time-to-market, minimal defects and post-design rework. These advantages lead to quantifiable benefits within the product development lifecycle, in terms of time and cost. Through one case study, this paper will provide Six Sigma, Process Excellence and Lean practitioners with the rationale for spreadsheet simulation and optimization in DFSS initiatives. Discussion topics include the role of simulation and optimization in the DMADV methodology, disadvantages of not quantifying uncertainty in DFSS projects, differences between deterministic and stochastic optimization, and tradeoff considerations when running optimizations. Practical techniques for efficiently identifying robust, high quality solutions are demonstrated through the use of Monte Carlo simulation and optimization.

## 1 INTRODUCTION

What is quality? From the customer perspective, product attributes that successfully fulfill their wants (e.g., function, features, colors and designs) would define quality. This is termed Customer Quality. From a manufacturer perspective, quality is the elimination of problems customers do not want (e.g., defects, failures, noise, vibration, unwanted phenomena, etc). This is termed Engineered Quality. Failure to consider both kinds of quality often leads to an unsuccessful product. Current research shows that about 45% of development resources result in killed or financially failed products (Brue and Launsby 2003), for reasons ranging from inadequate market analysis, to higher than anticipated costs, poor timing of introduction or quality production problems.

Failure can occur because of problems at any point in the product development life cycle (research, design, production, customer delivery). Yet most current Six Sigma efforts concentrate on the quality aspects of the production and customer delivery phases, where defects are easy to see, but costly to fix. Applying Six Sigma methods in the earlier phases of design and prototyping, where defects are harder to predict, but easier to fix, is the intent of the Design for Six Sigma (DFSS) methodology. This methodology has defined phases (Define, Measure, Analyze, Design, Validate) that many abbreviate as DMADV. DFSS in the planning stages of product development can save an estimated 15% on costs in later operations (Brue and Launsby 2003).

In this paper, we will explore how simulation and optimization can be integrated into the DFSS methodology to increase product development opportunities. Better product development yields many benefits, from faster time-to-market, higher quality, lower costs and more efficient production.

This paper is divided into two parts. The first section covers Monte Carlo simulation, using a practical case study to demonstrate the effectiveness of simulation as a DFSS tool. The second section addresses optimization. We reprise the case study from the simulation overview to outline optimization's role and effectiveness as a DFSS tool.

## 2 MODELS AND SIMULATION

Let's start with two definitions.

**Definition 1** *Models are an attempt to capture behavior and performance of business processes and products.*

**Definition 2** *Simulation is the application of models to predict future outcomes with both known and uncertain inputs.*

Models come in many different forms:

- Mathematical relationships based on established physical principles;

- Regression equations derived from historical data;
- Design of Experiments (DOE) response equations from measured observations;
- General knowledge of business system or product.

When one experiments with a model to measure the possible future behavior of a system (e.g., simulation), one gains insight into the nature of a process, identifies problems with system design and manage risk by making decisions with a better understanding of the costs and benefits.

One particular type of simulation, called Monte Carlo simulation, uses statistical sampling to approximate a solution to a problem. Stanislaw Ulam, a mathematician who worked on the Manhattan Project during World War II, is often credited with inventing this method.

Monte Carlo simulation can be described as a computer simulation of N trials where, for each trial, the algorithm:

- Samples input values from defined probability distribution functions (PDFs);
- Applies those sampled input values to the models and records the output; and
- Uses sampling statistics on the recorded output values to characterize the output variation (e.g., mean, standard deviation, fitted probability distributions).

The output of a Monte Carlo simulation has two important components. First, it predicts the output variation. A histogram, or forecast chart, shows the full range of possible outcomes and the probability of their occurrence. Second, it identifies the primary variation drivers. A sensitivity analysis pinpoints which few critical factors cause the predominance of variation in the response variable of interest.

By showing the output variation and identifying the primary drivers, Monte Carlo simulation aids the DFSS process in three key areas: product development cycle costs, product development cycle time and quality levels. The use of simulation can demonstrably reduce the cost of iterating and building sequential physical prototypes and the associated test costs, not to mention the time and resources expended going through design cycle iterations. Quality can be predicted and improved upon before making any single prototype or implementing transactional process changes. The following case study illustrates this conclusion.

For our case studies, we created our models with Microsoft® Excel, and used Crystal Ball® software (an Excel-based application) from Decisioneering®, Inc. to perform the simulation, optimization, analysis and reporting. One can download the Crystal Ball models from `<www.crystalball.com/wintersim/models.html>`. While Excel is sufficient to view the deterministic models,

one needs the Crystal Ball software version 7 or higher to run the simulations and optimizations.

## 2.1 Case Study: Liquid Packaging Pump

This case study demonstrates the role that simulation provides while designing a Liquid Packaging Pump. The pump will draw processed fluids from a reservoir into jars at a consistent rate, in order to keep up with consumer demand. The approach consists of identifying the optimum Flow Rate and allowable variation (upper and lower specification limits, or USL and LSL, respectively). Then, we will use known flow equations with Monte Carlo simulation on the first-cut design to determine the Flow Rate Mean Response, the Flow Rate Variation (Standard Deviation), the Flow Rate Defect Rate and the Design Parameters that most influence variation. There is a choice of nine available pump models and nine backflow valve models, each with its own cost and mechanical parameters. If one picks motor option number 3 and valve option number 9, what would be the estimated quality defect rate and part cost?

With the known data, we could build a physical prototype, measure and analyze the flow rates over a number of experiments and, through several iterations, finalize the pump design. This approach takes both time and money. Instead, we will create a virtual prototype (the model in Excel) and approximate the experiments in a fraction of the time with Monte Carlo simulation. Note that while Figure 1 shows a partial screen shot of the Excel model, it is highly recommended to download the model to view all the equations.

Marketing and Production considerations developed in the Design phase determine the acceptable range of flow rates:

1. How many units need to be produced daily to meet the market demand?
2. What is the useful production time per day, based on plant setup and productivity estimates?
3. What is the number of pumps per plant?

From these considerations, we derive the engineering specification limits for Flow Rate:

- LSL = 47.62 ml/sec;
- USL = 53.57 ml/sec;
- Target = 50.595 ml/sec.

The Flow Rate equation is known:

$$F = (K \pi R^2 L - B) S \qquad \text{where}$$

| | A | B | C | D | E | F | G | H | I | J | K |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | Nominal | | | StDev | | Component | |
| 14 | | | Component Option # | Lower Limit | Initial Value | Upper Limit | Lower Limit | Initial Value | Upper Limit | Cost Function | Sampled Value |
| 15 | | | | | | | | | | | |
| 16 | | | | | | | | | | | |
| 17 | | | | | | | | | | | |
| 18 | | Piston Radius (mm) | | 25.0000 | 30.0000 | 50.0000 | 0.10000 | 0.33333 | 0.33333 | $8.33 | 30.0000 |
| 19 | | Stroke Length (mm) | | 25.0000 | 35.0000 | 100.0000 | 0.10000 | 0.33333 | 0.33333 | $14.85 | 36.0000 |
| 20 | | Motor Speed (rpm) | 3 | 30.0 | 30.00000 | 40.0 | 0.10000 | 0.24000 | 0.24000 | $3.25 | 30.0000 |
| 21 | | Backflow (ml) | 9 | 0.10 | 0.50000 | 0.50 | 0.00250 | 0.01500 | 0.01000 | $0.30 | 0.5000 |
| 23 | | Target Flow Rate | | | 50.5950 | | | | FLOW RATE FORECAST (ml / sec) | | 50.6467 |
| 24 | | Target Flow Rate LSL | | | 47.2617 | | | | | | |
| 25 | | Target Flow Rate USL | | | 53.9283 | | | | TOTAL COST FORECAST ($) | | $26.73 |
| 27 | | COST COEFFICIENTS FOR RADIUS AND LENGTH OPTIONS | | | | | | | | | |
| 28 | | Base Cost Coefficient of Piston (radius) | | | 0.008 | | | | | | |
| 29 | | "Tolerance" Cost Coefficient of Piston (tolerance) | | | 1.25E-01 | | | | | | |
| 30 | | Base Cost Coefficient of Housing (length & radius) | | | 0.0004 | | | | | | |
| 31 | | "Tolerance" Cost Coeff of Housing (radius tolerance) | | | 1.25E-01 | | | | | | |
| 32 | | "Tolerance" Cost Coeff of Housing (length tolerance) | | | 1.25E-01 | | | | | | |
| 34 | | MOTOR AND VALVE OPTION COST AND RELATED STDEV'S | | | | | | | | | |
| 35 | | | | | option cost | option nominals | option stdev's | | | | |
| 36 | | | | | | | | | | | |
| 37 | | Motor Cost | Option #1 | | $3.75 | 30.0 | 0.10000 | | | | |
| | | | Option #2 | | $3.50 | 30.0 | 0.17000 | | | | |

Figure 1: Partial Screenshot of the Fluid Pump Excel Model

- F is Flow Rate (ml / sec);
- K is a Constant;
- R is Piston Radius (mm);
- L is Piston Stroke Length (mm);
- B is Backflow Valve (ml);
- S is Motor Speed (rpm).

The initial nominal values and standard deviations for the motor and backflow valve options were obtained from the potential suppliers, as well as their associated costs. The initial nominal values for the piston radius and stroke length were proposed by the designing engineers. Known machining capabilities for the piston radius and stroke length were determined during the Measure phase. Cost functions were also developed that link the raw material usage based on radius and length plus a term that increases the machining cost when requesting a smaller tolerance with similar capabilities (Creveling 1997).

We then created the Excel (or deterministic) model, representing the above system. If one has downloaded the models, use the spreadsheet titled DFSS Fluid Pump.xls. Figure 1 shows a partial screenshot of that same model. The deterministic model shows that the forecasted flow rate is 50.6467 ml/sec with a cost of $26.73.

During the Analyze phase, predictions of flow rate variation are provided using Monte Carlo simulations. The input values (piston radius, stroke length, motor speed and backflow) are assigned probability distributions (the Normal distribution in this example), based on the initial nominal and standard deviations (these are termed Assumptions within Crystal Ball). The user then specifies the number of Monte Carlo trials to run at; in this case, 10,000 trials. For each trial, the computer algorithms select different input values for each Assumption defined and the flow rate and total cost forecasts are calculated and saved. (This

is equivalent to the virtual building of 10,000 prototypes.) (The outputs measured are termed Forecasts within Crystal Ball.) The resulting plot of all 10,000 possible outcomes is shown in Figure 2.
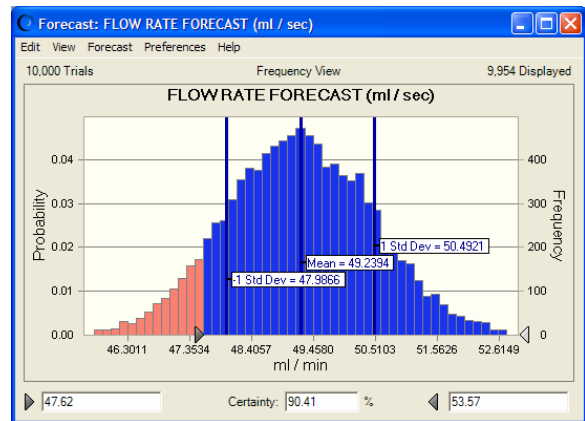


Figure 2: Flow Rate Forecast after 10,000 Trials

The nominal response rate of 49.24 ml/sec is close to the target (50.595 ml/sec), but 9.59% will fall out of the spec limits. This represents a Sigma Level of approximately 1.3.

Next, we ran a sensitivity analysis to determine the major drivers of the variation. Figure 3 shows the results.

The Piston Radius is the main driver. Can we reduce the standard deviation of the Radius? The designing engineers had already consulted with the machine shop and determined that tighter tolerances for the radius could be maintained but would require additional labor (more time per machining operation). The additional cost was reflected in the cost models.
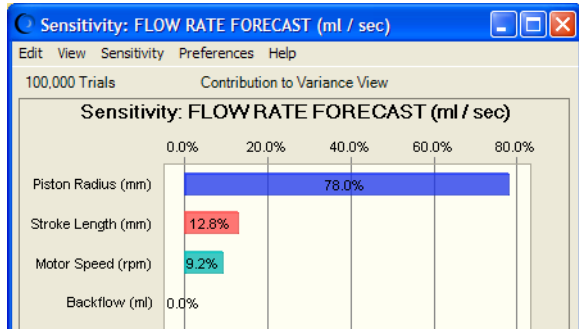
Figure 3: Results of the Sensitivity Analysis

Thus, reducing the standard deviation of the Piston Radius from 0.3333 to 0.1667 adds an extra $6.74 per part (total cost now of $33.47).

Running the Monte Carlo simulation with the reduced radius standard deviation results in a Sigma Level increase to 2.0, with 2.32% out of specification.

## 2.2 Case Study Conclusions

By modeling the system in a spreadsheet and applying Monte Carlo simulation, we were able to keep development time and costs low by eliminating the need to build physical prototypes at this stage of the project. We were also able to find a solution that increased the expected Quality Level (although perhaps not the optimal solution, see section 3.1). Plus, we were able to identify which of the variables had the most impact on the variability of the output.

## 3 OPTIMIZATION

What is optimization and when should we use it? We can use optimization in our work to achieve multiple goals (example: How can one minimize costs of production and also provide 6 Sigma quality levels?). We also, quite often without realizing it, use optimization in our everyday lives (example: What's the best time to leave the house if I want to miss the worst of rush hour and still be at my desk by 8:30am?).

Optimization, then, is an activity that aims at finding the best, or optimal, solution to a problem. We should use it when we have to weigh a decision amongst multiple objectives. The ideal time to utilize optimization is within the Design phase.

The goal of the optimization,i.e., finding the best solution, can be either single objective or multiple objective. To meet a single objective optimization goal, you determine a set of input values that either maximize or minimize a single objective function. For example: minimize product development cycle time, based on staffing levels. Intuitively, one realizes that there are many other goals that may conflict with the single objective approach. Therefore, multiple objective optimizations become more realis-

tic. For multiple objective optimizations, the goal is to determine a set of input values that influence multiple objective functions to their target values. An example of this would be to increase structural rigidity while reducing material usage (or reducing cost).

In addition to goals, one will need to consider constraints and requirements. Constraints apply to the inputs over which the designer has a degree of control (these are termed Decision Variables within Crystal Ball). The constraints are feasibility concerns expressed mathematically as boundaries that cannot be trespassed. Packaging dimension limits are good examples of constraints. Requirements are statistical goals to be achieved on the outputs (Forecasts) of interest. They are limits that are expressed mathematically as boundaries and represent the quality goals needed to achieve certain Sigma levels. For example, the time to complete the transaction should be less than five minutes.

Figure 4 provides a graphical representation of an optimization finding the "best" input ($X_{optimum}$) that maximizes the output ($Y_{max}$), subject to constraints on the input (input must be greater than $X_{C1}$ and less than $X_{C2}$) and a minimum requirement ($Y_{R1}$) on the output.
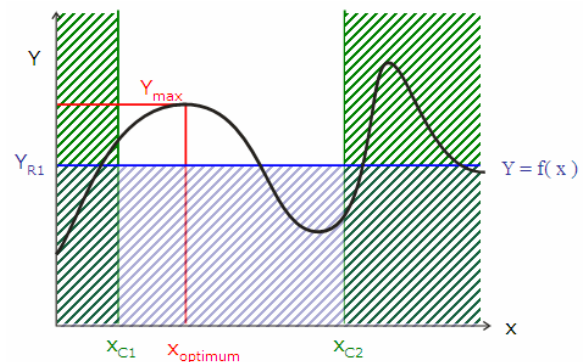


Figure 4: Constraints and Requirements

Different software applications can be used for optimization. Some, such as Microsoft® Excel Goal Seek and Solver, utilize local search algorithms. In a local search, the optimization search in the solution space (the allowed input values) is initiated at one location until the optimal output is achieved and cannot be improved upon using the algorithm. In a global search, as with OptQuest®, the optimization software packaged with Crystal Ball, the optimization search looks over the entire solution space, regardless of the initial search location.

As an analogy, imagine mountain climbing in the Rocky Mountains. In a local search, if one started somewhere on Pikes Peak, one would reach the summit (the maximum) at 14,110 feet and then stop. If all of the Rockies were clouded in, one would surmise this is the highest point within the Rockies. However, should the clouds lift as in a global search, the entire solution space (the Rocky

Mountains) would be visible and then the obvious high-point would be Mount Elbert at 14,433 feet.

Once one has determined whether to use local or global search methods, consideration is also needed for deterministic vs. stochastic optimization methods. A deterministic optimization identifies a single set of input values that place the output at its optimum.  It results in an optimal single-point estimate design.

Stochastic optimization adds statistical requirements on the outputs. Much as simulation allows for real-world variation to be accounted for in calculating the initial forecast (going from a single point estimate to a range of forecast values), stochastic optimization allows for variation to be accounted for while optimizing the design. The result is an optimal design where the output is no longer a single-point estimate, but rather falls within acceptable variation limits.  This enables the DFSS practitioner to implement Sigma quality level goals on any of the outputs.

Figure 5a provides a graphical representation of a value for the input ($X_{good}$), along with its natural variation ($\sigma_x$), that provides an output response that is both on target between the lower ($Y_{R1}$) and upper ($Y_{R2}$) requirement bounds, as well as results in an acceptable level of output variation outside those same bounds.  Contrast this with a value for the input value ($X_{bad}$, Figure 5b) that provides an output response that is near its target but does not result in an acceptable level of output variation.  A stochastic optimization program would identify $X_{good}$ as the "best" solution.

Typical optimization applications range from utilization of employees for workforce planning, configuration of machines for production scheduling, location of facilities for distribution, tolerances in manufacturing design, portfolio management, and many more.

The benefits of optimization in DFSS can be grouped into three main areas. First, optimization lets one consider multiple aspects of the problem. Multiple design constraints are specified up front while enabling multiple goals and requirements. Second, one improve the product development cycle time and cost.  One can rapidly search and evaluate many design possibilities with the use of automated search algorithms.  Knowledgeable decisions are made without having to build costly prototypes and replacing expensive testing iterations with optimization iterations. Third, stochastic optimization lets one include real-world effect  of input variation and allows simultaneous consideration of cost, performance and reliability in designing the best process or product.
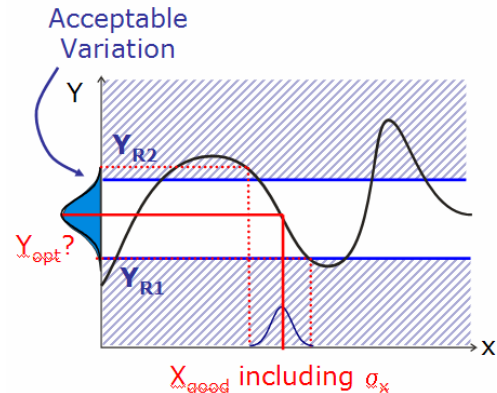


Figure 5a: Stochastic Requirement with Acceptable Output Variation
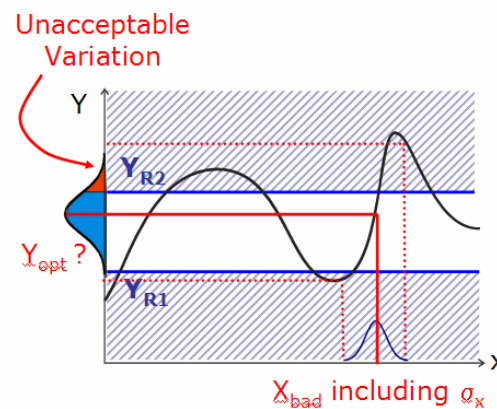


Figure 5b: Stochastic Requirement with Unacceptable Output Variation

## 3.1  Case Study Revisited: Liquid Packaging Pump

To demonstrate the role of optimization within the DMADV phases, we revisit the Liquid Packaging Pump case study.  In our earlier example, we wanted to design a pump to provide a flow rate between specification limits, accounting for variation around the inputs (or Assumptions). We found that the initial designs and Monte Carlo simulation increased predicted quality levels from 1.3 to 2.0 Sigma, but also with increased costs.

How can the pump design be configured so that minimal defects are produced for the least cost per part?  This is question that would need to be answered during the Design phase.

First we consider relationships between design parameters and cost. Choosing a certain radius has an effect on costs, as does selecting a certain stroke length, motor speed and backflow.  The cost relationships are represented in the model (see section 1 for instructions on downloading the models).

The cost response is a function of:

- Radius Mean ($\mu_R$);
- Radius Standard Deviation ($\sigma_R$);
- Stroke Length Mean ($\mu_L$);
- Stroke Length Standard Deviation ($\sigma_L$);
- Different Motor Options;
- Different Valve Options;
- Different cost coefficients (K) for the radius and length options.

The resulting cost equations are:

$$\$_{PISTONHEAD} = K_1 * \mu_R{}^2 + K_2 / \sigma_R{}^2$$

$$\$_{HOUSING} = K_3 * \mu_R{}^2 * \mu_L + K_4 / \sigma_R{}^2 + K_5 / \sigma_R{}^2 .$$

In setting up an optimization, one must first decide which variables the designer has ability to change or influence in the Design phase (termed Decision Variables). In Figure 1, or in any Crystal Ball model, these variables are in cells colored yellow. One also identifies the range at which the Decision Variables can vary between (the Constraints). In this example, the designing engineer determined raw stock availability with a supplier to define a range of piston radii and stroke lengths as possible nominal values, identified in Figure 6.

| Decision Variables | Lower Bound | Upper Bound | Variable Type? |
|---|---|---|---|
| Radius Mean | 25 | 50 | Continuous |
| Radius StDev | 0.1 | 0.3333 | Continuous |
| Length Mean | 25 | 100 | Continuous |
| Length StDev | 0.1 | 0.3333 | Continuous |
| Motor Option | 1 | 9 | Discrete |
| Valve Option | 1 | 9 | Discrete |

Figure 6: Range of Decision Variables for Radii and Lengths

Much as simulation picks different input values for each trial and saves the corresponding set of output values to create the forecast, optimization picks different decision values for each optimization run, and then runs a complete simulation, checks whether the results are optimal and feasible (meets the statistical requirements) and, if not, chooses a different set of values and runs the optimization again.

In our case study, we want to minimize the cost (the primary objective) subject to the requirement that we reduce the variation of the flow rate to 3 Sigma levels (i.e., 99.73% of outcomes fall between LSL of 47.62 and USL of 53.57 ml/sec).

Instead of manually (and randomly) choosing different combinations and noting if the next is better than the last,

we let the optimization methods perform the iterations. Figure 7 (Performance Graph) displays the cost objective values as the series of optimization iterations are performed. As more iterations are performed, lower cost alternatives are generated, all meeting the new 3 Sigma quality requirement. Eventually, the most promising configuration is identified and very little reduction in cost can be achieved with further iterations, resulting in a typical asymptotic behavior ("flattening out" of the Performance Graph curve). This new optimized design resulted in an increase to 3 Sigma quality and a $5.48 piece price reduction (total cost = $28.47).
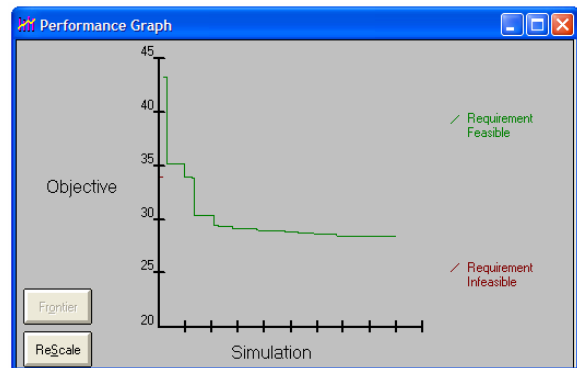


Figure 7: Optimized Design for Reduced Cost and 3 Sigma Quality Level

With just one optimization run, we discovered a balance with the right combination of decision variables that lets us pay less and have a higher Sigma Level than when we finished the earlier case study on simulation (2 Sigma and $33.47 vs. 3 Sigma and $28.48).

We can also quickly consider other designs. What if we wanted to achieve a 4-Sigma performance and minimize costs? Without having to build a prototype, we simply change the requirements in the software and run the optimization again. This time, we can achieve a 4-Sigma level and still only pay $33.37.

### 3.2 Case Study Conclusions

What has optimization allowed us to do in this case study? First, we were able to quickly consider multiple aspects of the problem. Further refinement was possible by changing the requirements, running the optimization again, and evaluating the recommended design configuration. Second, we were able to rapidly search and evaluate many design possibilities and make knowledgeable decisions without having to build costly prototypes. Third, stochastic optimization allowed us to include real-world variation of inputs for much more realistic design.

## 4 CONCLUSIONS

Design for Six Sigma provides a disciplined approach to product development, yielding great opportunities. Its advantages include increased competitiveness, increased profitability, reduced process and product development cycle time and increased quality.

In any design process, it is important to recognize that all inputs are susceptible to variation. While designing the product or process, one cannot know precisely how the system will behave at some future date. The ability to model input variation, predict output variation with the right tools and gain greater insight into the critical elements driving the response variation is a key step in a successful DFSS implementation.

Deterministic decision-making cannot predict the probability of failure or success; stochastic analysis can. Models with simulation and optimization that account for event probabilities are highly effective means to predict and improve business processes and products.

One caveat worth mentioning is that adoption of these techniques may require a change in organizational thinking. Changing from a single-point-estimate view of the world, to thinking in ranges and probabilities, can sometimes be challenging. Yet the successful application of these techniques throughout different business functions from executive to finance, engineering and operations will yield a true and lasting competitive advantage.

## REFERENCES

Brue, G., and R. Launsby. 2003. *Design for six sigma*. New York: McGraw-Hill.

Creveling, C. M. 1998. *Tolerance design*. Reading, Massachusetts: Addison-Wesley.

## AUTHOR BIOGRAPHIES

**KARL LUCE** is a Six Sigma Products Consultant and Master Black Belt at Decisioneering, Inc. His primary activities include the development and delivery of training courses for Decisioneering. Additionally, Karl delivers consulting services and support for the analysis, design, and implementation of spreadsheet models using Crystal Ball. Prior to his current role, he was Design for Six Sigma Master Black Belt at Lear Corporation where he mentored DFSS project teams, taught DFSS principles, and oversaw creation and confirmation of predictive models. He has applied DFSS on a wide range of automotive interior products such as Seats, Instrument Panels, Door Panels, and Headliners. His past assignments include managing CAE groups in Sweden and U.S.A. and serving as a Transactional Six Sigma Black Belt. He received a B.S. in Aeronautical/Astronautical Engineering from M.I.T. in 1985 and has a broad range of engineering knowledge within the aerospace and automotive industries. His e-mail address is <kluce@crystalball.com> and his Web address is <www.crystalball.com>.

**LUCIE TREPANIER** is Director of Marketing at Decisioneering, Inc. Her primary activities include marketing and product management for Crystal Ball software in the United States and internationally. Additionally, Lucie works closely with Product Development to define product direction. Prior to her current role, she was Senior Product Manager for Vericept Corporation (network security software), and Director of Product Management at Positron Inc. (telecommunications equipment), developing and marketing different product lines. She also was an electrical engineer, cost estimator and project manager for Tecsult, Inc., a large Canadian engineering consulting firm. She received a B.S in Electrical Engineering and an MBA, both from McGill University (Montreal, Canada). Her e-mail address is <ltrepanier@crystalball.com> and her Web address is <www.crystalball.com>.

**FRED CIOCHETTO** is a Trainer, Product Consultant and Six Sigma Product Manager. He is a Six Sigma Master Black Belt and holds a MS in Operations Research and a BS in Mathematics from the Colorado School of Mines. Prior to joining Decisioneering in 2003, Fred held managerial positions within Fortune 500 corporations in the Mining, Oil & Gas, Communications, Technology and Management Consulting sectors. He has extensive experience in delivering substantial and measurable business results through application of Lean Speed and Six Sigma Quality. He is particularly interested in risk analysis, stochastic modeling and optimization techniques for driving business process improvement. His e-mail address is <fciochetto@crystalball.com> and his Web address is <www.crystalball.com>.

**LAWRENCE GOLDMAN** is Director of Six Sigma Marketing at Decisioneering in Denver, CO. He joined the company in June 1997 as a program manager with the product development group and has held positions as webmaster, web producer, trainer, and newsletter editor. His current focus is educational and training materials and seminars for Crystal Ball. Prior to working for Decisioneering, Lawrence was employed as a trainer and technical support assistant for Techbase, a company specializing in mining and engineering database software. He has an M.S. in geology from the University of Cincinnati and a B.A. in geology from Cornell University. His email address is <lgoldman@crystalball.com> and his Web address is <www.crystalball.com>.