Quality Technology &
Quantitative Management
Vol. 3, No. 2, pp. 161-177, 2006

QTQM

Achieving Robust Design from Computer Simulations

Ron A. Bates[1], Ron S. Kenett[2], David M. Steinberg[3] and Henry P. Wynn[4]

[1, 4] London School of Economics, London, UK
[2] KPA Ltd., Raanana, ISRAEL
[3] Tel Aviv University, Tel Aviv, ISRAEL and KPA Ltd.
(*Received* December 2004, *accepted* July 2005)

Abstract: Computer simulations are widely used during product development. In particular, computer experiments are often conducted in order to optimize both product and process performance while respecting constraints that may be imposed. Several methods for achieving robust design in this context are described and compared with the aid of a simple example problem. The methods presented compare classical as well as modern approaches and introduce the idea of a 'stochastic response' to aid the search for robust solutions. Emphasis is placed on the efficiency of each method with respect to computational cost and the ability to formulate objectives that encapsulate the notion of robustness.

Keywords: Computer experiments, emulators, kriging, optimization, quality engineering, Taguchi.

1. Introduction

Product development is a fundamental aspect of long-term business survival ([28], [30]). In today's competitive, global economy, it is crucial to reduce the time required for design and development in order to reach the market more quickly, and there is a great need for techniques that can accelerate this. In this article we concentrate on the use of computer simulators as a platform for performing experiments in product and/or process development. Computer simulations are used to study a wide range of phenomena, from automobile performance to chemical mixtures. The idea is to replace actual physical experimentation in the laboratory or test bed with a "virtual" experiment in which a computer runs a program that simulates the behavior of the true process. As Thomke observed ([26]), experiments can be run on the simulator at a much faster rate than in the laboratory, allowing companies to dramatically reduce the time, cost and effort needed to conduct physical experiments. Moreover, process settings that might have been thought risky or even potentially dangerous can easily be tested. When physical testing is destructive, computer simulations are much more economical than actual experiments. See [27] for an example of the use of simulators to study the crashworthiness of automobiles at BMW. However, larges numbers of simulations can be very costly to perform, as single simulation runs may require many hours of CPU time to compute. In this case computer experiments may be employed to build fast approximate models of the simulator. These models are variously referred to as emulators, surrogates or meta-models and can be used to reduce the overall computational burden of a detailed analysis of the design by combining various levels of simulation fidelity with expert opinions and physical experiments (see [1], [9], [11], [12], [15] and [16]).

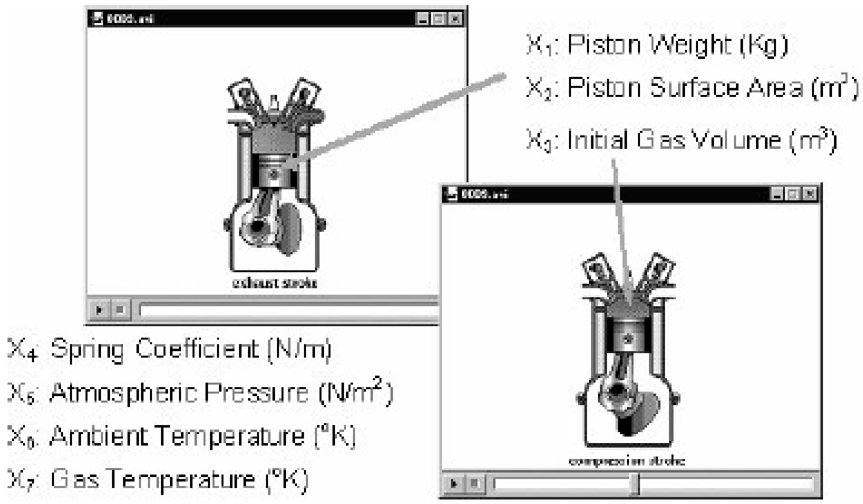

Figure 1. The piston simulator with seven design factors.

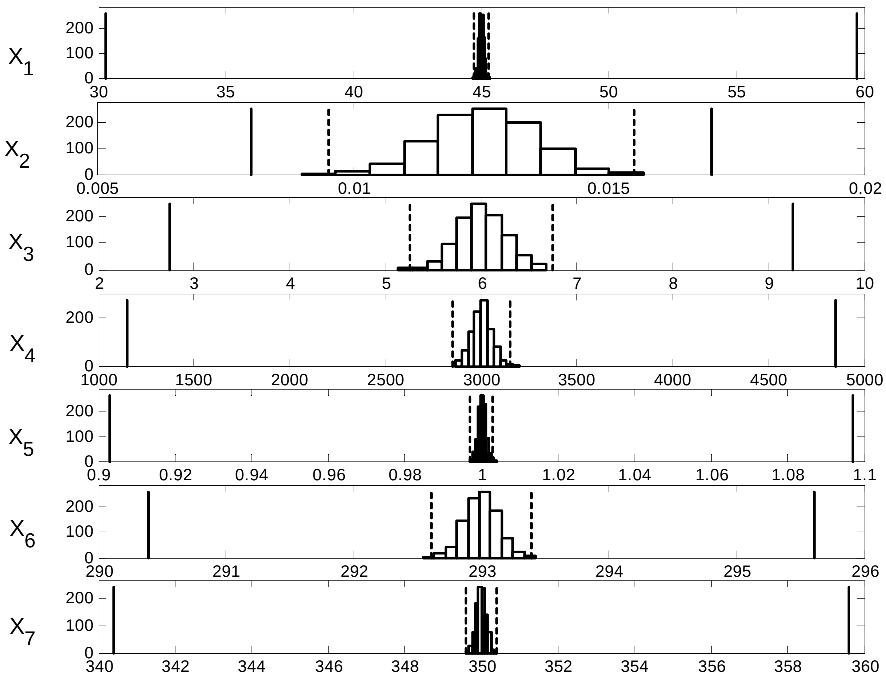

Figure 2. Design factor distributions with mean values set to nominal levels. Dashed lines represent ±3 standard deviations from nominal. Solid lines represent the effective limit on nominal factor values once noise is taken into account.

Competitive products must also meet strict standards for quality and reliability. In this area, robust design experiments are an important quality engineering tool for developing low-cost, yet high quality products and processes. A primary goal of these experiments is to reduce variation and the name "robust design" derives from the idea of making products insensitive, or "robust", to the effects of natural variations in their production and use environments. Robust design was pioneered by Genichi Taguchi ([24]) in Japan and has been embraced by many engineers around the world in the last 20 years. For more detail on robust design see [14] and [21]. Kenett and Zacks ([10]) include special chapters with a modern treatment of classical design and analysis of experiments, robust design and reliability analysis.

One of the fundamental concepts in robust design experiments is the separation of factors into "design factors" and "noise factors". Design factors are factors whose levels are determined within the engineering specification of the process. Noise factors are factors that cannot be set precisely during full-scale production or product use. A noise factor might reflect varying field conditions, variability in raw materials characteristics or the variation of a design factor about its nominally specified value. A robust design computer experiment can study process variation by simulating the variance transmitted to important output variables by the variations in the design factors and the use environment.

We will present and compare different strategies for designing and analyzing robust design computer experiments. These strategies involve the use of several methods including crossed array designs (inner and outer array), Signal to Noise ratios, orthogonal arrays, factorial analysis, space-filling experimental designs and spatial modeling. We present a new approach, the stochastic emulator method, in which we construct a statistical model (emulator) of the computer simulator and then use the emulator to assess performance characteristics. The next section provides details of the simulation model that will be used throughout the paper. The following sections include a description of the methods and details of the results achieved with each method. We then compare the methods for a particular robust design task and conclude with a discussion and final remarks.

2. The Piston Simulator

The piston simulator used in this paper is taken from Appendix II of Kenett and Zacks ([10]). The simulator's response variable is cycle time of a complete revolution of the piston's shaft, denoted by Y. Regulation of the piston's performance can be achieved by changing seven design factors: Piston weight (X_1), Piston surface area (X_2), Initial gas volume (X_3) Spring coefficient (X_4), Atmospheric pressure (X_5), Ambient temperature (X_6) and Filling gas temperature (X_7), see Figure 1. Table 1 shows the permissible range for each factor. The actual values of the factors exhibit normally distributed variation about the nominal levels, which results in variation in the cycle time, Y. The standard deviation, $\sigma(X_i)$, i=1,...,7, of each factor is also given in Table 1. Figure 2 shows the variation in each factor value relative to its range when the nominal level is set at the mid-point of the range. The effective minimum and maximum factor values are illustrated by solid lines in the Figure, and show that the search for a robust set of inputs is restricted to values that are at least 3 standard deviations (dashed lines) from the original limits. This is so that the addition of noise will not take a factor's value outside its range.

3. Robustness method

In this section we briefly describe several alternative methods that have been proposed for achieving robust design. Throughout we will denote the inputs to the simulator by

Table 1. Range of permissible factor levels and standard deviations, $\sigma(X)$ used by the piston simulator.

Factor	Design Label	Noise Label	Units	Minimum	Maximum	$\sigma(X)$
Weight	X_1	N_1	Kg	30	60	0.1
Surface Area	X_2	N_2	m^2	.005	.02	0.001
Initial Gas Volume	X_3	N_3	m^3	.002	.01	0.00025
Spring Coefficient	X_4	N_4	N/m	1000	5000	50
Atmospheric Pressure	X_5	N_5	N/ m^2	90000	110000	100
Ambient Temperature	X_6	N_6	K^0	290	296	0.13
Filling Gas Temperature	X_7	N_7	K^0	340	360	0.13

$X_1,\ldots,X_k$ and will focus on a single output Y. Take, for example, a simulation of a car crashing into a highway barrier. Some of the inputs will be design factors (e.g. the thickness of an auto bumper or the material from which it is made) whereas other factors may be essentially random (e.g. the angle of impact with a barrier). The common goal of robustness methods is to find nominal settings of the design factors for which a desired output distribution is obtained. The output goals are often to obtain a given mean value with small variance, but more general objectives might also be relevant (see, for example [13]).

Each one of the studies' methods was used to optimize the piston simulator with the objective of achieving a mean cycle time, $\mu(Y) = 0.20$ seconds while minimizing the standard deviation, $\sigma(Y)$. Note that, throughout the paper, sample estimates of the mean and standard deviation of Y will be referred to as m(Y) and s(Y), respectively. Table 4 summarizes the results. In order to obtain a fair comparison, each method used the same number of runs of the piston simulator because in the design of more complex products this is usually the most expensive and time-consuming aspect.

4. Taguchi-Style Design and Analysis

Taguchi ([24] and [25]) recommends using 'crossed array' experimental plans that include both design factors and any controllable noise factors. Two separate arrays are constructed, an inner array for the design factors and an outer array for the noise factors. These are "crossed" so that the simulator is run once for each combination of design and noise factor settings in the plans. When the factor is both a design factor (nominal level) and a noise factor (tolerance), the joint levels determine the final input value used by the simulator. Suppose, for example, the inner array called for a nominal piston weight of 3200 grams and the outer array for a tolerance of 80 grams below the nominal. Then the corresponding simulator run would use a weight of 3120 (= 3200 – 80) grams. For the design factors, Taguchi recommended using orthogonal arrays with most factors at either two or three levels. The outer arrays are typically two-level fractional factorials with the levels set at $\pm\sigma(X_i)$, i=1,...,k.

In the piston example, each of the 7 factors participates as both a design factor, reflecting possible choices of nominal values, and as a noise factor, reflecting manufacturing tolerances. The L8 experimental plan shown in Table 2 was used for both inner and outer arrays so that a total of 64 runs were required. As only two nominal levels are used for each factor, it is prudent not to extend the factors to the extreme limits of their operating range. In this experiment, the low and high levels used for each factor were chosen 1/6 and 5/6 of the way, respectively, from the minimum to the maximum values permitted (the experimental levels are encoded as ±1). The tolerance shifts from these fixed levels were $\pm 2\sigma(X_i)$. The tested design factor combinations are shown in Table 2, along with the mean response, m(Y), standard deviation of response, s(Y), and the signal-to-noise (SN) ratio. The overall mean cycle time was 0.456. The regression coefficients indicate the effect of each factor on the SN ratio and the mean and these are listed in Table 3.

The Taguchi analysis indicates that setting X_2, X_3 and X_6 to their high levels is the best way to control variation. The best factor for then adjusting the mean level appears to be X_4, which has the strongest effect on the mean among factors with low effect on SN. Note that X_2 and X_3 are also the strongest factors for the mean, so that adjusting them to reduce variation might make it difficult to get the mean on target.

Table 2. The design factor combinations in the Taguchi experiment, along with the mean, m(Y), standard deviation s(Y) and signal-to-noise ratio (SN) of the cycle times.

X_1	X_2	X_3	X_4	X_5	X_6	X_7	m(Y)	s(Y)	SN
-1	-1	-1	-1	-1	-1	-1	0.505	0.108	30.82
1	-1	-1	1	1	-1	1	0.444	0.098	30.23
-1	1	-1	1	-1	1	1	0.193	0.024	42.03
1	1	-1	-1	1	1	-1	0.324	0.040	41.71
-1	-1	1	-1	1	1	1	0.707	0.094	40.24
1	-1	1	1	-1	1	-1	0.627	0.048	51.38
-1	1	1	1	1	-1	-1	0.320	0.029	48.17
1	1	1	-1	-1	-1	1	0.530	0.052	46.47

Table 3. Effect of the design factors on the signal-to-noise ratio (SN) and the mean cycle time, m(Y), according to the Taguchi analysis.

Factor	Effect on SN	Effect on m(Y)
X_1: Weight	1.07	0.025
X_2: Surface Area	3.22	- 0.115
X_3: Initial Gas Volume	5.18	0.090
X_4: Spring Coefficient	1.57	- 0.060
X_5: Atmospheric Pressure	- 1.30	- 0.008
X_6: Ambient Temperature	2.46	0.007
X_7: Filling Gas Temperature	- 1.64	0.012

5. Crossed Array Design with Response Model Analysis

The analysis of data from crossed array robust design experiments can also take explicit account of noise factors ([22], [23]). The idea here is to carry out a statistical analysis of the entire matrix of data from a crossed-array experiment, including main effects for the noise factors and interactions between design and noise factors. This approach has come to be known as response model analysis, by contrast with the performance measure analysis advocated by Taguchi. The former directly analyzes the responses from the simulator whereas the latter first computes performance measures (SN and m(Y)) that summarize the responses at each design point.

One of the important ideas in the response model analysis is that the variance transmitted to the output by the noise factors is related to the design by noise factor interactions. Suppose there is an interaction between the noise factor N_1 and the design factor X_1. The interaction implies that the regression slope of Y on N_1 is $\alpha_{1+}\alpha_{1,1}X_1$ and depends on the level of X_1, which can then be chosen to make the slope closer to 0. The transmitted variance is proportional to the square of the slope, so the nominal level of X_1 can be chosen to help make the product robust to variations in N_1.

The experiment is exactly the same as in the previous section. We now carry out a conventional factorial analysis, fitting main effects for all design and noise factors and all design by noise two-factor interactions. Figure 3 is a half-normal probability plot of the factor effects. The plot shows that X_2 and X_3, as design factors, have the strongest effects on Y. The most important contributor to variation is N_2, which has the largest standard deviation of all the factors relative to its viable range. The plot also highlights an interaction effect of X_2 with N_2, therefore setting X_2 to its "high" level from the parameter design experiment reduces the variance generated by N_2. There is also an X_4N_2 interaction (not labeled in Figure 3). The average coefficient for N_2 is -0.053, but with X_2 and X_4 set to their high level, the estimated coefficient is -0.018, only about 1/3 as large as the average value. The next largest contributor to variation in Y is N_3. There is a moderately large interaction between X_3 and N_3 (also not labeled in the Figure 3). The latter interaction reflects the fact that the effect of X_3 on Y is nonlinear, with the slope steeper at lower values and flatter at higher values. Setting X_3 to its high nominal level exploits the flat part of the curve to reduce the variance transmitted by N_3. The slope of N_3 is reduced from 0.019 (on the average) to 0.010 by setting X_3 to its high nominal level.

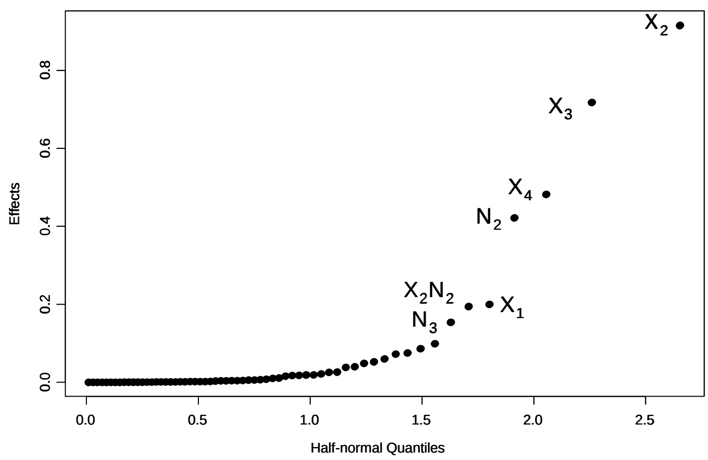

Figure 3. Half-normal probability plot of factor effects for a response model analysis of a parameter design experiment on the piston simulator.

6. Dual Response Surface Analysis

The dual response surface approach [29] provides an immediate and direct way to assess jointly how the mean and dispersion of Y depend on the design factors. An experimental plan is prepared such that each experimental point gives nominal values for the design factors. Replicate measurements are generated by the simulator at each design point and are summarized by their mean and a measure of dispersion, such as s(Y), or its logarithm. A crossed array design is recommended to obtain the replicates, but Vining and Myers ([29]) also noted that the replicates could be sampled at random from the distributions specified for the noise factors. Finally statistical models are built to relate the design factors to the mean and dispersion of Y. This strategy was initially proposed in the context of physical experiments with replication, but it can be applied equally well to computer experiments.

For the piston example, two different experimental plans were considered: a 32 run Latin Hypercube Sampling design (LHS) with two replicates, and a 2^{7-3} fractional factorial design with four replicates, again giving 64 function calls in each case. The replicates were sampled at random from the specified tolerance distributions. Then m(Y) and s(Y) were computed at each design point and used to construct the models.

The LHS design has experimental settings that cover the range specified in Figure 2 for each design factor. At the 32 experimental settings, m(Y) ranged from 0.16 to 1.01 with a mean of 0.46 and s(Y) ranged from 0.005 to 0.096 with a mean of 0.024. A nonlinear kriging model, which is a spatial model specially developed for use with computer experiments [18], [19], was fitted to m(Y). Figure 4 shows a Generalized (leave-one-out) Cross Validation (GCV) plot and a main effects plot for the fitted model. The main effects plot indicates the individual effects of each factor in the model, with the effects of other factors removed. It shows clear negative main effects for X_2 and X_4, and positive main effects for X_3 and X_1. The remaining three factors had less effect on the average cycle time. The effect of X_2 was found to be mildly nonlinear, with a steeper slope at low values. The GCV results show that the overall fit was good, with a Root Mean Squared Error equal to 3.89% of the overall range in the response. Expressing the RMSE values as a percentage of the range of the response gives a useful indication of the accuracy of the model.

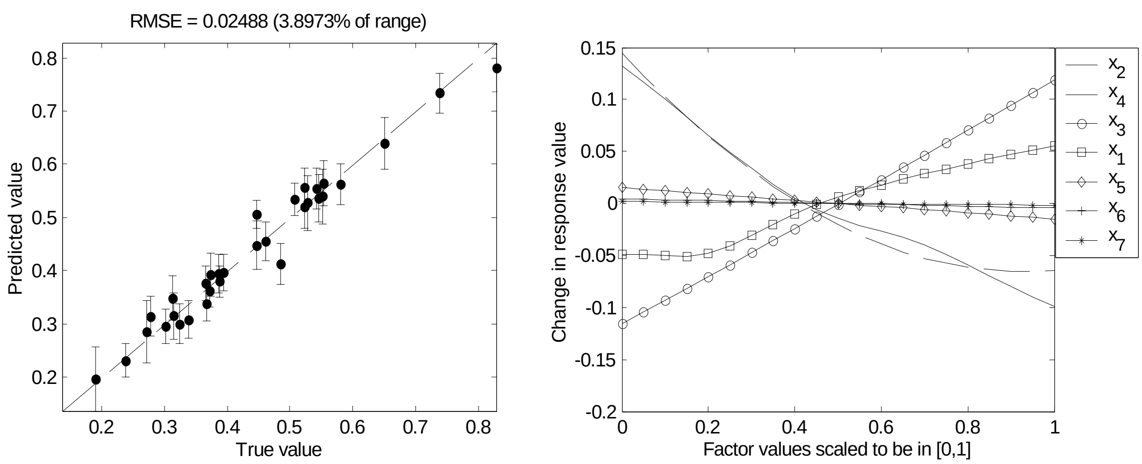

Figure 4. Results for LHS kriging model of mean cycle time, m(Y). Left: cross validation with 95% confidence intervals, right: main effects.

A kriging model was also fitted to s(Y), shown in Figure 5, but had poorer predictive ability, with a GCV root mean squared error equal to 16% of the overall range. Four factors had a significant effect in the s(Y) model, with X_2 and X_3 the most important. X_5 and X_6 had weaker effects. The effects of all four factors were estimated to be nonlinear.

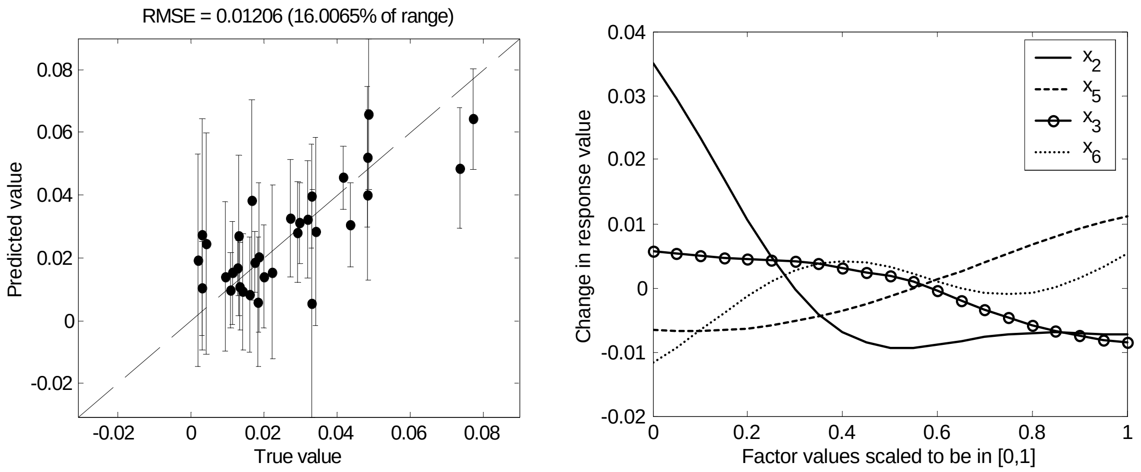

Figure 5. Results for LHS kriging model of standard deviation of cycle time,s(Y). Left: cross validation with 95% confidence intervals, right: main effects.

For the 2^{7-3} fractional factorial plan, the nominal levels for each factor were set at 1/6 and 5/6 of the way from the low end of the range to the high end, as in the cross-product array. A standard aliasing scheme was used, with *E=ABC*, *F=ABD* and *G=ACD*. Here, m(Y) ranged from 0.18 to 0.93 with a mean of 0.46 and s(Y) from 0.010 to 0.042 with a mean of 0.030. Conventional factorial analyses were conducted on both m(Y) and s(Y) and Figures 6 and 7 present half-normal probability plots of the effects. The strongest effects on m(Y) were due to X_2 and X_3, with a somewhat weaker effect for X_4. These conclusions are identical to those found earlier from the cross-product array. Figure 7 fails to identify any strong effects on s(Y). However, the strongest effect, due to X_2, has a coefficient about 40% of the mean of s(Y), suggesting that it might be possible to achieve an important reduction in variation by increasing X_2. Further reduction may be possible by decreasing the nominal value of X_3, due to the interaction of this factor with X_2. (Actually there are three pairs of factors aliased with this contrast, but one might guess that this pair is the most likely, given the strong main effect of X_2.) The same conclusions result if log(s(Y)) is considered in the factorial analysis.

With the two designs used here, the dual response analysis had difficulty identifying factor effects for s(Y). The designs pointed to X_2 as being the most likely factor to influence s(Y), but the results were not conclusive. The analyses of m(Y) for both designs also pointed to X_2 as the most important factor. The LHS design picked up the nonlinearity in this effect whereas the fractional factorial has no ability to do so.

7. The Stochastic Emulator Strategy

The first step in implementing the stochastic emulator strategy is to run an experiment using an array that includes both design and noise factors. Research to date on computer experiments has recommended "space filling" designs such as Latin Hypercube Sampling (LHS) designs or lattice designs rather than orthogonal arrays or fractional factorials, in order to achieve more uniform coverage of the input space. See [3], [5], [7], [18], [19] for more detail on possible designs. The emulator can be of any model type such as kriging estimators ([18]), radial basis functions ([6]) also see Bates ([2]) for an engineering application), polynomial functions ([4]) or adaptive regression splines ([8]).

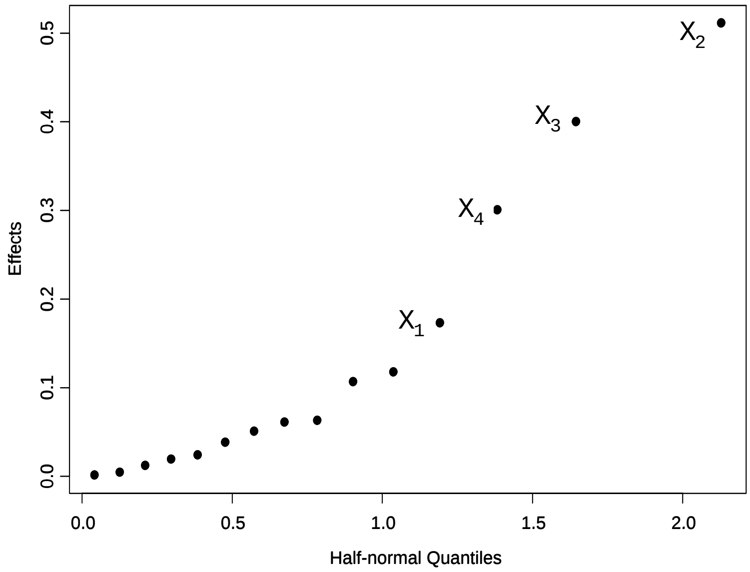

Figure 6. Half-normal probability plot of the factor effects on mean cycle time, m(Y) in the 2^{7-3} fractional factorial design.

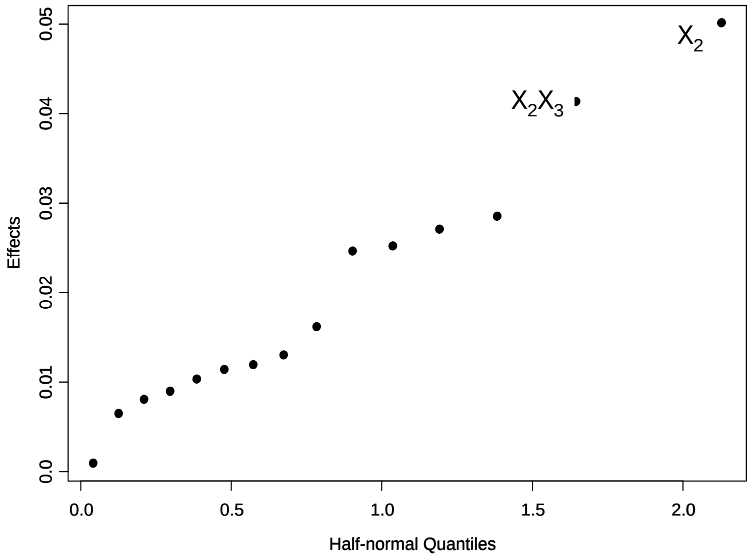

Figure 7. Half-normal probability plot of the factor effects on the standard deviation of cycle time, s(Y) in the 2^{7-3} fractional factorial design.

In this case we first seek to build an empirical model (an emulator) that represents the relationship between all factors and the chosen response. No distinction is made at this point between the dual roles of the factors as design factors (for setting nominal values) and as noise factors (representing deviation from nominal). For a given set of factor values the emulator estimates the piston cycle time, Y, and can also be used to estimate the effect of noise on Y by studying how it behaves when subject to small changes in factor values.

For the 7-factor piston example, the initial design was an LHS design with 64 points. The lower and upper bounds for the design factors were taken to be the same as those given in Table 1. The response Y was then evaluated at each point using the piston simulator. The emulator, chosen here to be a kriging model ([18]), was then fitted to the data. Leave-one-out cross validation results can be seen in Figure 8 (left-hand diagram), which shows the true (simulator) vs. predicted (emulator) responses, including 95% confidence intervals for the predictions, along with GCV RMSE calculations expressed, as before, as a percentage of the range of the response. A value of 0.96% indicates that the emulator is accurately estimating the simulator response.

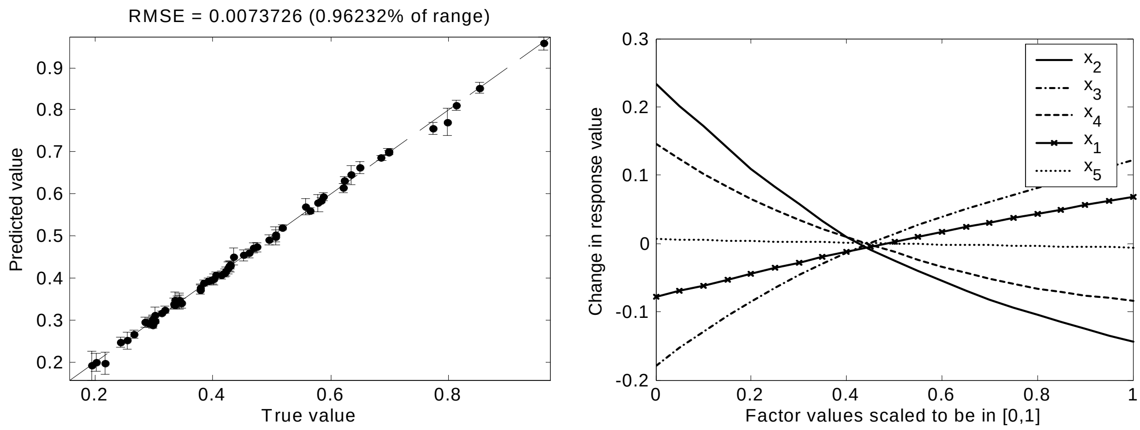

Figure 8. Fitted stochastic emulator of mean cycle time m(Y) using LHS design. Left: cross validation with 95% confidence intervals, right: main effects.

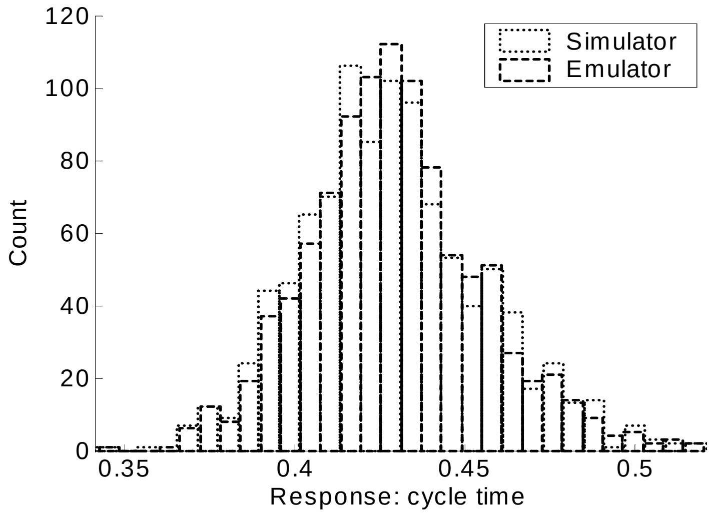

Figure 9. Histogram of Piston response when all design factors are set to their mid-range. The dotted lines show the cycle time distribution from the simulator, the dashed lines the distribution from the emulator.

Additionally, a main effects plot (Figure 8 right-hand diagram) shows the estimated effect that each factor has on Y. The Figure shows that X_2 and X_3 have the strongest main effects, whereas X_6 and X_7 had negligible main effects and were not included. The emulator can now replace the simulator to assess the effect of noise on Y for any nominal input values. A large sample of points is generated from the tolerance distributions and Y is estimated from the emulator. The excellent fit illustrated in Figure 8 indicates that the emulated distribution should be very close to the actual noise distribution for the simulator. Figure 9 shows two histograms of 200 responses generated by the emulator and the simulator when all the inputs are set at their mid-ranges and then allowed to vary randomly according to the noise distributions defined in Table 1.

Nominal settings of the design factors can now be chosen by studying a variety of features of the stochastic response histogram, including:

(1) Mean, m(Y)

(2) Standard deviation, s(Y)

(3) Mass of estimated response distribution within an interval

(4) Mass of estimated response distribution above or below a threshold level

The search for nominal factor settings can be approached in several ways, depending on the objective function.

(1) Random search. Randomly generate sets of factor values, use the emulator to estimate the output distribution, calculate the stochastic response and choose the best set of values.

(2) Experimentation. Generate a second space-filling design (e.g. a LHS design) and calculate the stochastic response at each design point. At this stage there are two options:

   i. Choose the best design point.

   ii. Build a second emulator to model how the input factors affect the stochastic response and optimize this model.

We repeat here, as described in Figure 2, that the search space of the stochastic emulator is reduced by $\pm 3\sigma(X_i)$, so that the first emulator remains valid during the search. Monte Carlo analysis is used to estimate the response distribution at input factor values chosen according to a 128 point LHS design. At each point in the LHS design, a 200-point random sample was generated according to the specified noise distributions. This sample was then used to estimate both mean and variance at each LHS design point. Two stochastic emulators were then built to estimate m(Y) and s(Y). For s(Y), the fitted stochastic emulator GCV RMSE was 0.0014, equivalent to 3.90% of the range of the response. This is shown in Figure 10 (left-hand diagram) along with the 95% confidence intervals for the prediction. The right-hand diagram of Figure 10 shows the main effects plot of the s(Y) stochastic emulator.

The m(Y) stochastic emulator is almost identical to the initial emulator shown in Figure 8, as one would expect, with a GCV RMSE of 0.0036, equivalent to 0.66% of the range of the response. This emulator is required because, in order to combine mean and variance, the emulators need to be based on the same sample data. A constrained optimization was then performed, minimizing the s(Y) stochastic emulator, while requiring that the m(Y) stochastic emulator satisfy the constraint of equality to 0.2.

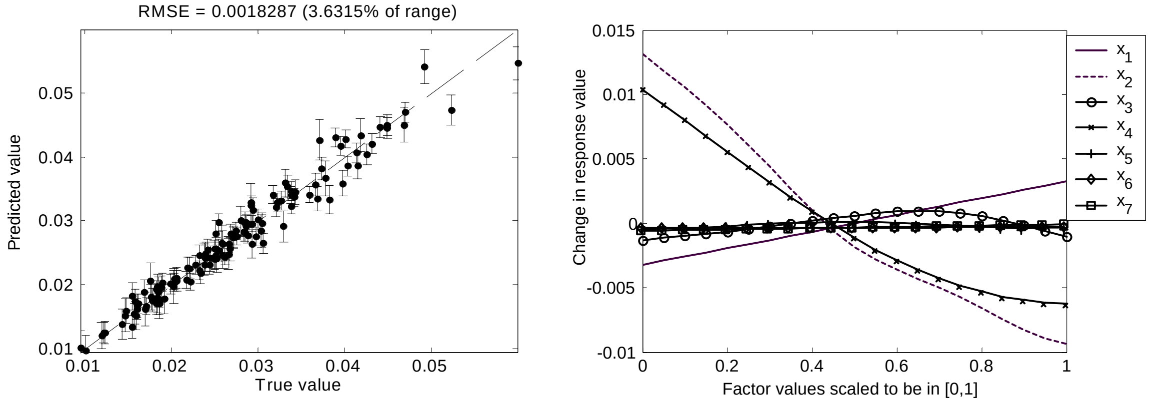

Figure 10. Fitted stochastic emulator of standard deviation of cycle time, s(Y) using LHS design. Left: cross validation with 95% confidence intervals, right: main effects.

8. Comparison of Results for Each Strategy

Table 4 presents recommended factor settings derived from each of the analyses methods along with the results of 1000 actual simulator runs at those conditions. We begin with a brief explanation of how the recommended settings were obtained. The Taguchi analysis indicated that setting X_2 to a high level is beneficial for both m(Y) and s(Y) and setting X_6 to its high level has almost no effect on the mean. However, increasing X_3 to its high level moves m(Y) well above 0.2, even if X_1 is set to its minimal search level and X_4 to its maximal search level. A small amount of trial and error shows that, to achieve an estimated mean of about 0.2, the X_3 should be set to 0.0044, about half-way between its minimum and mid-range. The factors X_5 and X_7, which have minimal effects, were set arbitrarily to their mid-ranges.

The recommendations from the response model analysis for reducing variation involved the three design factors with the strongest effects on m(Y) (see Table 3). As with the Taguchi analysis, setting all three factors to high levels made it impossible to achieve the desired average cycle time of m(Y)=0.2 seconds. In particular, it was necessary to adopt a lower nominal level for X_3, despite the harmful implications this had for s(Y). The only difference between the two analyses is that the response model analysis did not find any important effect for X_6, which could be set arbitrarily to its mid-range.

Table 4. Collected results for all methods for the problem of minimizing cycle time standard deviation s(Y), subject to a target mean value of m(Y)=0.2 seconds.

	X_1	X_2	X_3	X_4	X_5	X_6	X_7	m(Y)	s(Y)
Taguchi	30.3	0.017	0.0044	4850	100,000	295.6	350	0.204	0.0106
Response Model	30.3	0.017	0.0044	4850	100,000	293	350	0.204	0.0110
2^{7-3} Dual Response	45	0.017	0.00275	3426	100,000	293	350	0.218	0.0137
	45	0.017	0.00488	4850	100,000	293	350	0.263	0.0137
	45	0.017	0.00382	4138	100,000	293	350	0.242	0.0132
LHS Dual Response	34.5	0.014	0.00346	4245	109,700	295.61	355.9	0.230	0.0154
Stochastic Emulator I	30.3	0.017	0.00359	3924	101,610	294.5	340.4	0.199	0.0107

The dual response surface designs were less successful than the crossed array designs in identifying factor settings that provide a value of m(Y)=0.20 with a small s(Y). The analysis of the 2^{7-3} design favored high settings for X_2, which reduce the value of both m(Y) and s(Y). As no other effects were found on s(Y), the remaining factors could be set to adjust m(Y) to the target value. Numerous solutions were possible. Table 4 shows three such solutions, each based on using the next strongest factors, X_3 and X_4, to adjust m(Y). The first solution set X_3 to its extreme search setting and then adjusted X_4, the second reversed the roles of the two factors, and the third used intermediate settings of both factors. All of the proposed solutions had values for m(Y) that were more than one standard deviation above the target value. In addition, the standard deviations were about 25% larger than those achieved in the earlier analyses. For the 32-run LHS design, we used the kriging models as inputs to an optimizer, with the goal of minimizing s(Y) subject to m(Y)=0.20. The resulting solution is shown in Table 4. Although these factor settings had excellent predicted performance (based on the kriging models), their actual performance was not good, with m(Y) more than 2.5 standard deviations off target.

For the stochastic emulator approach, the recommended factor settings and the results from actually running the simulator are shown in Table 4. The histogram of the response distribution at this setting is shown in Figure 11, which also shows an excellent match between the emulator and simulator results.

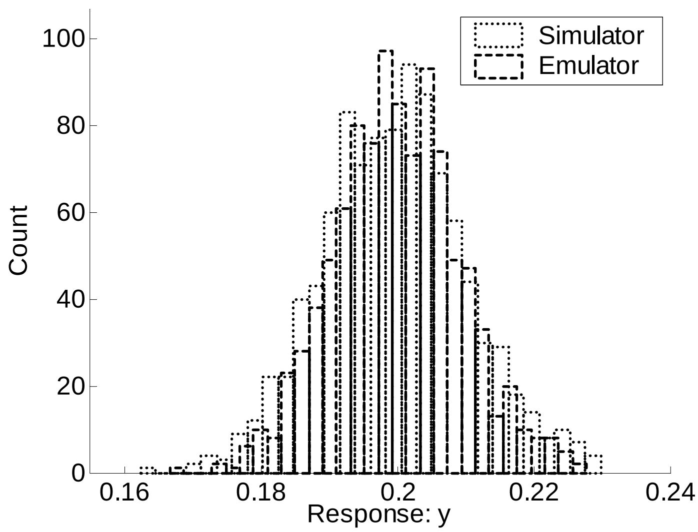

Figure 11. Histogram of Piston response (200 run Monte Carlo simulation) with all design factors set to their optimal values. The dotted lines show the cycle time distribution from the simulator, the dashed lines the distribution from the emulator.

For the piston example, the Taguchi method, the response model analysis and the stochastic emulator all provided better solutions than the dual response method. In particular, they did a much better job of keeping the mean cycle time, m(Y), on target. Our implementation of the dual response method estimated m(Y) at nominal input values by a small random sample of results drawn from the tolerance distribution about the nominal values. The results suggest that small random samples are much less effective here than the systematic tolerance sampling in the crossed array design or the large random samples used with the stochastic emulator.

9. Discussion

Several Robust Design methods have been described and applied to the problem of setting piston cycle time to a pre-specified target value with a minimal standard deviation. The results showed a clear preference for the methods that modeled the mean cycle time using systematic or large random samples. The dual response surface analyses, which used average cycle times from small random samples about the design points, were much less effective in modeling the mean cycle time, even with the use of larger designs for nominal settings, than the crossed array. As a result, the cycle time distributions achieved with the dual response surface method were shifted off-target by 1 to 4.5 standard deviations. The other methods produced distributions centered much closer to the target cycle time.

Although the methods led to similar standard deviations in the cycle time distribution, the dual response surface method, with random samples, also proved less able to discern which factors affect cycle time standard deviation. This result is not surprising, as it parallels the analysis of Steinberg and Bursztyn ([23]), who showed that control of noise factors and directly modeling their effects greatly heightens the power to detect dispersion effects in physical experiments.

It is also interesting that the Taguchi approach produced good results for this example. The small number of trade-off decisions and the ability to reduce cycle time standard deviation with some design factors and adjust to target with others meant that a good solution was obtainable in a manual fashion. It is easy to see that this approach might not be so successful in a more complex case where there are many more design decisions to be made.

The stochastic emulator approach is a novel and potentially very useful tool in robust design for computer experiments. The advantage of this approach is that experimental effort is focused on producing an accurate model of the simulator. This is more likely to produce better models than can be found with, say, crossed arrays. The initial emulator is then used to build stochastic emulators that produced an excellent robust design solution for our trial problem. Moreover, this approach provides a general framework to attack more complex problems related to the output distribution, such as maximizing the probability mass in an interval or to one side of a threshold.

We end with a cautionary note. From a quality standpoint, robustness is an important aspect in judging product performance. For example, there is good reason to prefer a design that reduces sensitivity to noise factors or that reduces the fraction of outliers in system performance. Our robustness analyses are thus aimed at optimizing objective functions that reflect robustness. From a narrow mathematical point of view, then, we succeed in finding "optimal" settings. However, from a broader perspective, it is essential to remember that robustness requires lack of sensitivity to both expected, and unexpected, variations. Our simulators, and to a still greater extent, emulators of simulators, reflect only expected variations. Yet it may be the unexpected, surprise variation that is the main source of serious performance failures. One potential pitfall in using mathematical optimization tools is that they often push the system design to extreme settings and thus actually reduce robustness to unexpected variations. When using these tools, every effort must be made to avoid this type of over-engineering.

Acknowledgement

This research was supported by the European Union grant TITOSIM (Time to Market via Statistical Information Management, project number: GRD1-2000-25724).

References

1. Allen, T., Bernshteyn, M. and Kabiri-Bamoradian, K. (2003). Constructing meta-models for computer experiments. *Journal of Quality Technology*, 35, pp. 264-274.
2. Bates, R.A. and Wynn, H.P. (2000). *Adaptive Radial Basis Function Emulators for Robust Design, in Evolutionary Design and Manufacture*, I C Parmee (Ed.), 343-350, Springer Verlag.
3. Bates, R.A., Buck, R.J., Riccomagno, E. and Wynn, H.P. (1996). Experimental design and observation for large systems. *Journal of the Royal Statistical Society, Series B*, 58, 77-94.
4. Bates, R.A., Giglio, B., and Wynn, H.P. (2003). A global selection procedure for polynomial interpolators. *Technometrics*, 45, 246-255.
5. Bates, R.A., Kenett, R.S., Steinberg D.M and Wynn, H.P. (2004). Robust design using computer experiments, *Proceedings of the 13-th European Conference on Mathematics for Industry (ECMI)*, 21-25 June 2004, Eindhoven, the Netherlands , Springer Verlag.
6. Broomhead, D.S. and Lowe, D. (1988). Multivariate functional interpolation and adaptive networks. *Complex Systems*, 2, 321-355.
7. Bursztyn, D. and Steinberg, D.M. (2005). Comparison of designs for computer experiments. *Journal of Statistical Planning and Inference*, in press.
8. Friedman, J.H. (1991). Multivariate adaptive regression splines. *Annals of Statistics*, 19, 1-141.
9. Huang D., Allen T. (2005), Design and analysis of variable fidelity experimentation applied to engine valve heat treatment process design. *Journal of the Royal Statistical Society, Series C – Applied Statistics*, 54, pp. 443-463.
10. Kenett, R.S. and Zacks, S. (1998). *Modern Industrial Statistics: Design and Control of Quality and Reliability*, Duxbury Press, San Francisco, CA.
11. Kennedy, M.C. and O'Hagan, A. (2001). Bayesian calibration of computer models. *Journal of the Royal Statistical Society, Series B*, 63, 425-464.
12. Li, R.L. and Sudjanto, A. (2005). Analysis of computer experiments using penalized likelihood in Gaussian Kriging models. *Technometrics*, 47, 111-120.
13. Moorhead, P.R. and Wu, C.F.J. (1998). Cost-driven parameter design. *Technometrics*, 40, 111-119.
14. Phadke, M.S. (1989). *Quality Engineering Using Robust Design*, Prentice-Hall, New York, NY.
15. Reese C., Wilson A., Hamada M., Martz H., Ryan K. (2004), Integrated Analysis of Computer and Physical Experiments. *Technometrics*, 46, pp. 153-164.
16. Romano, D. and Vicario, G., (2002), Reliable Estimation in Computer Experiments on Finite Element Codes. *Quality Engineering*, Vol.14, Issue 2.
17. Sacks, J. and Welch, W.J. (1991). A system for quality improvement via computer experiments. *Communications in Statistics--Theory and Methods*, 20:477-495.
18. Sacks, J., Welch, W.J., Mitchell, T.J. and Wynn, H.P. (1989). Design and analysis of computer experiments. *Statistical Science*, 4, 409-435.
19. Santner, T.J., Williams, B.J. and Notz, W.L. (2004). *The Design and Analysis of Computer Experiments*, Springer, New York, NY.
20. Shoemaker, A. C., Tsui, K.-L. and Wu, C.F.J.(1991). Economical Experimentation Methods for Robust Parameter Design. *Technometrics*, vol. 33, pp. 415-428.

21. Steinberg, D.M. (1996). Robust design: Experiments for improving quality, in *Handbook of Statistics*, Vol. 13, ed. by S. Ghosh and C.R. Rao, Elsevier Science, 199-240.

22. Steinberg, D.M. and Bursztyn, D. (1994). Dispersion effects in robust design experiments with noise factors. *Journal of Quality Technology*, 26, 12-20.

23. Steinberg, D.M. and Bursztyn, D. (1998). Noise factors, dispersion effects and robust design. *Statistica Sinica*, 8, 67-85.

24. Taguchi, G. (1986). *Introduction to Quality Engineering*, Kraus International Publications, White Plains, NY.

25. Taguchi, G. (1978). *System of Experimental Design*. Unipub/Kraus International Publications and American Supplier Institute 1978.

26. Thomke, S. (2001). Enlightened experimentation: The new imperative for innovation. *Harvard Business Review*, February, 67-75.

27. Thomke, S., Holzner, M., and Gholami, T. (1999). The crash in the machine. *Scientific American*, 280, 72-77.

28. Urban, G.L. and Hauser, J.R. (1993). *Design and Marketing of New Products*, Prentice-Hall, Englewood Cliffs, NJ.

29. Vining G G and Myers R H (1990). Combining Taguchi And Response-Surface Philosophies - A Dual Response Approach. *Journal of Quality Technology* 22 (1): 38-45.

30. Wheelwright, S.C. and Clark, K.B. (1992). *Revolutionizing Product Development*. The Free Press, New York, NY.

Authors' Biographies:

Ron Bates is a Senior Research Fellow in the Department of Statistics at the London School of Economics and Political Science (LSE), London, UK. He has worked on several EU-funded research contracts related to Robust Engineering Design and Product Development and is a consultant in Industrial Statistics for Enterprise LSE. He has a PhD in Robust Engineering Design from City University, London and a BSc in Engineering from Sussex University. He has published over 30 research and conference papers and his research interests include Design of Experiments, multivariate spatial modelling, time-frequency analysis and modelling complex nonlinear engineering and biomedical systems.

Ron S. Kenett has a Ph.D. degree. He is CEO and Senior Partner of *KPA Ltd.*, an international management consulting firm and Visiting Professor at the University of Torino, Torino, Italy. He has over 25 years of experience in restructuring and improving the competitive position of organizations by integrating statistical methods, process analysis, supporting technologies and modern human resource management systems. As Professor of Management at the State University of New York- Binghamton (1987-1993), he was awarded the General Electric Quality Management Fellowship. For ten years he served as Director of Statistical Methods for Tadiran Telecommunications Corporation and, previously, as researcher at Bell Laboratories in New Jersey and faculty member at the department of Statistics, University of Wisconsin-Madison. His 90 publications are on topics in industrial statistics, biostatistics and quality management. Ron is co-author of four books including - Modern Industrial Statistics: Design and Control of Quality and Reliability (with S. Zacks), Duxbury Press, 1998, Chinese edition, 2004, Multivariate Quality Control: Theory and Applications (with C. Fuchs), Marcel Dekker Inc., 1998 and Software Process Quality: Management and Control (with E. Baker), Marcel Dekker Inc., 1999. Professor Kenett's Ph.D. is in Mathematics from the Weizmann Institute of Science (1978); he earned a BSc. in mathematics with first class honors from Imperial College, London University (1974). Ron is president elect of ENBIS, the European Network for Business and Industrial Statistics (www.enbis.org). He can be reached at ron@kpa.co.il. More information on his publications is available at www.kpa.co.il.

David Steinberg is on the faculty of the Department of Statistics and Operations Research of Tel Aviv University. He also is head of the statistical consulting laboratory at Tel Aviv University. He completed his Ph.D. degree in Statistics at the University of Wisconsin-Madison. His main areas of research are experimental design, robust design and computer experiments. He has taught numerous courses indesign of experiments in academia and in industry. He is involved in many applications in a wide range of fields, including industrial projects, biostatisticsand seismology. His more than 70 published papers reflect this broad spectrum of statistical interest. He is an associate editor for Technometrics and has also served as an associate editor for Applied Statistics. David is also the Vice President of the Israel Statistical Association.

Henry Wynn is Professor of Statistics and Convenor (head) of the Statistics department at the London School of Economics. Previous to that he held senior positions at Imperial College, London, City University, London and the University of Warwick. Until recently he was also part-time Scientific co-Director of EURANDOM in the Netherlands. His extensive track record in theoretical and applied statistics includes working in experimental design, optimisation and robust engineering design. Industrial work includes European-union funded multi-disciplinary research contracts and direct research and consultancy for industry. He is an honorary fellow of several international societies.