QUALITY AND RELIABILITY ENGINEERING INTERNATIONAL
Qual. Reliab. Engng. Int. 2009; **25**:515–524
Published online 18 June 2009 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/qre.1044

Research

Design and Analysis for the Gaussian Process Model[‡]

Bradley Jones[1,*,†] and Rachel T. Johnson[2]

[1] *SAS Institute, SAS Campus Drive, Building S, Cary, NC 27513, U.S.A.*
[2] *Naval Postgraduate School, 1 University Circle, Monterey, CA 93943, U.S.A.*

In an effort to speed the development of new products and processes, many companies are turning to computer simulations to avoid the time and expense of building prototypes. These computer simulations are often complex, taking hours to complete one run. If there are many variables affecting the results of the simulation, then it makes sense to design an experiment to gain the most information possible from a limited number of computer simulation runs. The researcher can use the results of these runs to build a surrogate model of the computer simulation model. The absence of noise is the key difference between computer simulation experiments and experiments in the real world. Since there is no variability in the results of computer experiments, optimal designs, which are based on reducing the variance of some statistic, have questionable utility. Replication, usually a 'good thing', is clearly undesirable in computer experiments. Thus, a new approach to experimentation is necessary. Published in 2009 by John Wiley & Sons, Ltd.

KEY WORDS: surrogate model; computer experiments; space-filling designs; Latin hypercube design

1. INTRODUCTION

'In the 21st century computer simulation experiments will replace physical experiments for many applications.' With this provocative assertion, Professor Jeff Wu began his keynote address at an international conference on statistics in Tianjin, China in the summer of 2006. At first blush this seems like hyperbole for emphasis. But there are several reasons why it may not be as much an overstatement as it seems. For one, as quality, reliability, and productivity improvement efforts move into the design phase, the use of computer models becomes more attractive. This is because in design phase the creation of prototypes can be expensive and time consuming. Using computer models avoids the necessity of building a large number of prototypes. Another reason is that some systems do not allow for physical experimentation. For example, many people would be outraged by experimentation with hospital procedures in an intensive care unit. As another example, the national economy is an important system to understand, yet deliberate factorial experimentation would be infeasible. In both of the above cases, a computer simulation model could provide substantial insight.

A short list of current applications of computer simulation models includes circuit simulation, stress analysis testing, hurricane tracking, and turbulent flow studies. Currin *et al.*[1] describe an integrated circuit

[*] Correspondence to: Bradley Jones, SAS Institute, SAS Campus Drive, Building S, Cary, NC 27513, U.S.A.
[†] E-mail: Bradley.Jones@jmp.com
[‡] This article is a U.S. Government work and is in the public domain in the U.S.A.

simulation that was also presented and studied in Currin *et al.*[2] and Sacks *et al.*[3]. Allen *et al.* [4] describe a finite element analysis (FEA) model used for designing an 'interference fit' plastic seal for which the response of interest was the insertion force needed to fully engage the tongue with the groove. Modeling of short-term and long-term weather is another important application area for computer simulation. Several papers on the modeling and analysis of computer hurricane models include Johnson and Watson[5], Iman *et al.*[6], and Watson and Johnson[7]. Xiao *et al.*[8] use computational fluid dynamics (CFD) models to simulate turbulent mixing in jet engines; these models are used to predict turbulent mixing properties in a physical experiment.

Similar to physical experiments, the researcher performs a computer experiment by making a number of systematic changes to the parameters of a computer simulation model of the system under study. Computer experiments provide several advantages over physical experimentation. Computer experiments only require the programming of the model and are limited only by the speed of the processor(s). Prototypes used for physical experimentation are generally expensive and require substantial time to build. Computer experiments are comparatively cheap, only involving the cost of a computer and the time it takes to build the model. However, the one disadvantage of the computer experiments is their questionable ability to accurately predict the real world. Physical experiments have empirical validity, but whether a computer model is an adequate surrogate for the real system is an important consideration.

Current research addresses the calibration, verification, and validation of computer experiments via sophisticated statistical techniques. These three processes are critical to the delivery of computer simulation models that are adequate for predicting the behavior of real processes.

Model calibration deals with the issue of choosing the model parameters of a computer code so that the physical data are well approximated by the computer model. This problem is addressed in Park[9], Craig *et al.*[10], and Kennedy and O'Hagan[11]. Discrepancies between the output results from the computer model and physical experiments are analyzed and the computer model is often tuned until the measure of discrepancy is at or below a specified value. There are several different metrics for discrepancy. Examples can be found in Trucano *et al.*[12] and Pilch *et al.*[13].

While model validation and verification are often combined, we prefer to define them separately. Model verification involves comparing computer-generated output with a theoretical mathematical analysis of the system. Model validation aims to determine the degree to which data from a computer model approximate the data generated from the physical system it is representing. Model verification is done either prior to the model validation or iteratively with model validation. Roache[14], Hills and Trucano[15], and Pilch *et al.*[13] all present detailed discussion of various methods and strategies used for model verification.

Model validation is key to the development of a computer model. Bayarri *et al.* [16] recommend a framework for the validation of computer models. In this paper the authors state that the most important question in evaluation of a computer model is whether or not the computer model adequately represents reality. A model that adequately represents reality is one that provides sufficiently accurate predictions for the intended use. The authors also state that, 'in practice, the processes of computer model development and validation often occur in concert; aspects of validation interact with and feed back to development; for example, a shortcoming in the model uncovered during the validation process may require a change in the mathematical implementation.' While this phase of the model development is crucial, there are many issues surrounding the validation process as discussed in Berk *et al.*[17]. Sometimes it may be impossible to collect physical data. In these cases, alternate methods for validation must be used as described in Berk *et al.*[17], Sacks *et al.* [18], and Santner *et al.*[19]

Assuming that a computer simulation has been calibrated, verified, and validated, it can be used to make predictions about the behavior of the physical system it models. The use of designed experiments is equally important when studying a computer simulation as it is for understanding physical systems. Properly planned experimentation on a computer simulation is necessitated by the complexity of the underlying model. Computer simulations, while cheap compared with physical experiments, can often have very long run times. Allen *et al.*[4] point out that, 'even though FEA is intended to reduce costs compared with physical experimentation, finite element experiments are often time consuming and costly.' Moreover, computer simulations often depend on many variables. Exploring a multidimensional factor space requires efficient experimentation. For graphical exploration in this multidimensional space it is convenient to develop a simpler surrogate for the computer simulation model. A useful surrogate model is a closed-form mathematical expression that relates the input variables to the output response. Using a surrogate model of the computer simulation model allows for very fast (microsecond) predictions of new responses at design points not yet tested. The surrogate model then provides a cheap alternative to running the computer code, which may take hours or days. However, using a surrogate model requires verification that it provides adequate approximations of the computer simulation model.

Design augmentation is a way to provide this verification. Design augmentation involves adding runs to a design to gain extra knowledge. One typical goal of augmentation is to locate an optimum response. Another is to reduce the uncertainty of prediction in a region of interest. Design augmentation is common in physical experimentation, but there is limited literature discussing how to augment designs for deterministic computer simulation. Sacks *et al.*[3] discuss a sequential design algorithm for the Gaussian Process integrated mean-square error (IMSE) criterion. They mention the theoretical hardships associated with this technique and describe an algorithm that overcomes some of the pitfalls. Johnson *et al.*[20] discuss a design augmentation technique for space-filling designs used to fit high-order polynomials. They demonstrate the effectiveness of this design augmentation technique with respect to the prediction variance properties of the polynomial design.

The Gaussian Process model is a surrogate model that is widely used in the computer simulation research. In this paper, we describe the Gaussian Process model and present an example of its use. In Section 2, we discuss the Gaussian Process model and its prediction properties. In Section 3, we discuss designs used for computer simulation models and their pros and cons. In Section 4, we demonstrate an application of the design and analysis of a computer simulation via a test function. In Section 5, we present conclusions.

2. THE GAUSSIAN PROCESS MODEL

Gaussian Process (GASP) models are a currently popular choice for use as a surrogate for computer simulation models. They are flexible, meaning that they can fit a wide variety of surfaces, from very simple to quite complex. They are parsimonious in the number of parameters. For the correlation structure we use, the GASP model has only as many parameters as variables in the model plus a parameter to estimate the mean and another to estimate variance. A desirable feature of these models is that there is no error for input settings where you have observed responses. That is, GASP models interpolate the data.

Sacks *et al.*[3] proposed the GASP model for use as a surrogate for simulation output results. The GASP model is a statistical model adopted from the spatial statistics literature. For a specific choice of the correlation function, the Gaussian Process model is the kriging model[21]. The GASP model treats the deterministic output response as a realization of a random stochastic process; specifically a multivariate normal. The output response is represented as an $n\times 1$ data vector $y(x)$ with mean $\mu 1_n$ (also $n\times 1$) and covariance

$$Var(y)=\sigma^2 R(X,\theta)$$

where $R(X,\theta)$ is an $n\times n$ correlation matrix. The correlation matrix $R(X,\theta)$ is a function of the design space, design points, and unknown thetas. There are a variety of forms that can be used for this correlation function (see Sacks *et al.*[3]), but we will use the following form:

$$R_{ij}(X,\theta)=\exp\left(-\sum_k \theta_k (x_{ik}-x_{jk})^2\right)$$

where $\theta_k \geq 0$. If $\theta_k=0$, then the correlation is 1.0 across the range of the kth factor and the fitted surface is flat in that direction. Large θ_k correspond to low correlation in the kth factor and the fitted surface will be very bumpy (or wiggly) in that direction. The parameters μ, σ, and θ may be fit using maximum likelihood. If we represent the maximum likelihood estimates by $\hat{\mu}$, $\hat{\sigma}$, and $\hat{\theta}$, the prediction equation is

$$\hat{y}(x)=\hat{\mu}+r'(x,\hat{\theta})R^{-1}(X,\hat{\theta})(y-\hat{\mu}1_n)$$

DOI: 10.1002/qre

where $r_i(x, \hat{\theta})$ is an $n \times 1$ vector of estimated correlations of the unobserved $y(x)$ at a new value of the explanatory variables with the observations in the data, $y(x)$. The form of $r_i(x, \hat{\theta})$ is

$$r_i(x, \hat{\theta}) = \exp\left\{-\sum_{k=1}^{p} \theta_k (x_k - x_{jk})^2\right\}$$

Note that r has same form as the correlation matrix. If we replace the vector x above with the matrix X of the data, then r becomes R, which cancels with R^{-1}. Hence, $\hat{y}(x)$ equals y and thus the GASP model interpolates the data.

For the GASP model, the relative prediction variance discounting error in estimating parameters is

$$\frac{Var(\hat{y}(x))}{\sigma^2} = 1 - r'(x, \hat{\boldsymbol{\theta}}) R^{-1}(\mathbf{X}, \hat{\boldsymbol{\theta}}) r(x, \hat{\boldsymbol{\theta}}) + \frac{(1 - \mathbf{1}' R^{-1}(\mathbf{X}, \hat{\boldsymbol{\theta}}) r(x, \hat{\boldsymbol{\theta}}))^2}{\mathbf{1}' R^{-1}(\mathbf{X}, \hat{\boldsymbol{\theta}}) \mathbf{1}}$$

Once results are available, the prediction equation and the variance of prediction are useful for building plots of the predictions as a function of each simulation parameter.

3. DESIGN OF EXPERIMENTS FOR COMPUTER SIMULATIONS

Computer experiments are different from physical experiments in that they have no random error and they deal with functions that are thought to have more complex behavior than can be adequately modeled by a low-order polynomial. As a result of these differences, the design approach for computer experiments cannot use the same principles that guide the scientist in creating the physical experiments.

Because there is no random error involved in computer experiments, replication is not desirable. In fact, replicated points cause the correlation matrix, introduced in the previous section, to become singular. Hence, replication for GASP models creates problems. Because the underlying system is deterministic, randomization and blocking are also no longer useful.

For physical experiments, low-order polynomial functions are generally adequate for model fitting and response optimization. But for computer experiments, low-order polynomial approximation may not be adequate. Hence, traditional designs based on minimizing variance with respect to lower-order Taylor series expansions may not be useful since there is no variance to minimize.

These inherent differences between computer and physical experiments have led to the development of families of experimental designs specifically for use in computer modeling. These designs are known as space-filling designs. Examples of space-filling designs include the sphere packing design, the Latin Hypercube design, the uniform design, the maximum entropy design, and the GASP IMSE design.

The sphere packing design, or maximin design, maximizes the minimum distance between pairs of designs points. This design was developed in Johnson *et al.* [22]. The sphere packing design maximizes the minimum inter-site distance and is specified by

$$\max_{D} \min_{u,v \in D} d(u, v) = \min_{u,v \in D_*} d(u, v)$$

where $d(u, v)$ is a distance that is greater than or equal to zero, and D represents the design points. Examples of applications of the maximin designs can be found in Jank and Shmueli[23], Liefvendahl and Stocki [24], Chen *et al.*[25], Roux *et al.*[26], and Bursztyn and Steinberg[27]. Sphere packing designs are impressive at filling space, but their drawback is that they do not maintain uniform spacing when projecting into a lower-dimensional space. This is a desirable property for computer experiments because, as in physical experiments, many factors often prove to be unnecessary. Removing these factors from the design is the same as projection. If, as a result of this projection, two design points fall on top of each other, the result is pseudo-replication and the GASP model for this smaller number of factors will have a singular correlation matrix.

The Latin hypercube design was developed by McKay *et al.*[28]. It is defined in Fang *et al.*[29] as, ‘A Latin hypercube design (LHD) with n runs and s input variables, denoted by LHD(n,S), is an $n \times s$ matrix, in

DOI: 10.1002/qre

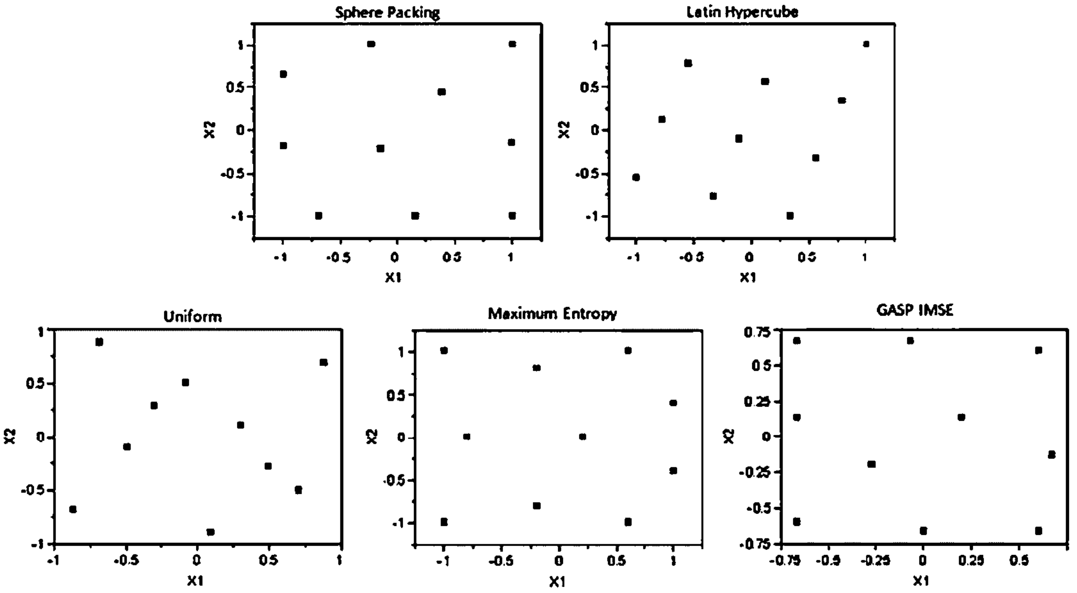


Figure 1. Examples of 2 factors, 10 run designs for 5 space-filling designs

which each column is a random permutation of $\{1,2,\ldots,n\}$.' Examples of applications of LHDs can be found in Bayarri *et al.*[16], Welch *et al.*[30], Mease and Bingham[31], Tyre *et al.*[32], and Storlie and Helton[33]. The Latin hypercube designs are by far the most widely used. Because of their construction they project uniformly onto each factor so pseudo-replication is never a problem. However, these designs do not fill the space as well as the sphere packing design.

The uniform design was created by Fang[34] and Wang and Fang[35]. The goal of uniform design is to find the set of points that most closely approximates a continuous uniform distribution. A measure of how close a given set of points comes to a perfect approximation is discrepancy. For a given number of points, the optimal uniform design minimizes the discrepancy. Fang[29] defines discrepancy. Let $F(x)$ be a uniform distribution on C^s (the unit cube) and $F_{Dn}(x)$ is the empirical distribution of the design D_n,

$$F_{D_n}(x)=\frac{1}{n}\sum_{k=1}^{n}\{x_{k1}\leq x_1,\ldots,x_{ks}\leq x_s\}$$

where $x=(x_1,\ldots,x_s)$ and $I\{A\}=1$ if A occurs, or 0 otherwise. In this paper, the software used to create designs displayed in Figure 1 uses the centered L_2 discrepancy found in Hickernell[36]. The L_2 discrepancy can be treated as an objective function which can be minimized in continuous space. An example of the application of a uniform design is found in Bursztyn and Steinberg[27]. Uniform designs are clearly useful for approximating the integral of an arbitrary function. It remains unclear how well they perform when the function to be fit is the GASP model.

The maximum entropy design, developed in Shewry and Wynn[37], uses entropy as the optimality criterion. Entropy is a measure of the amount of information contained in a data set. They show that the expected change in information is maximized by the design D that maximizes the entropy of the observed responses at the points in the design. If the data are assumed to be from a normal $(m,\sigma^2 R)$ distribution, where R is

$$R_{ij}=\mathrm{e}^{\left(-\sum_k \theta(x_{ik}-x_{jk})^2\right)}$$

which is the correlation of responses at two design points, then the design maximizes the determinant of $R(|R|)$ (Sacks *et al.*[3]). An application of this design can be found in Ko *et al.*[38]. The maximum entropy design

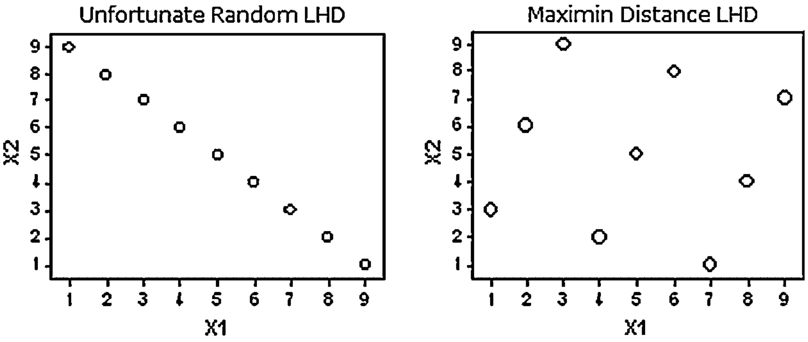


Figure 2. Example of two LHDs demonstrating the need for a secondary criterion

is analogous to the D-optimal design for GASP models. Applications of these designs are not widespread, but they could be useful because of the direct connection of their objective function to the GASP model.

The GASP IMSE was proposed by Sacks *et al.*[3]. The IMSE criterion chooses the design to minimize

$$\int_k MSE[\hat{y}(x)]\,\mathrm{d}x$$

where $MSE[\hat{y}(x)]$ is given by the prediction variance equation shown in Section 2. Like the maximum entropy designs, these designs have an objective criterion that is directly connected to the GASP model to be fit. One problem with these designs is that as the number of factors increases, they require substantially more points to do an adequate job of filling space.

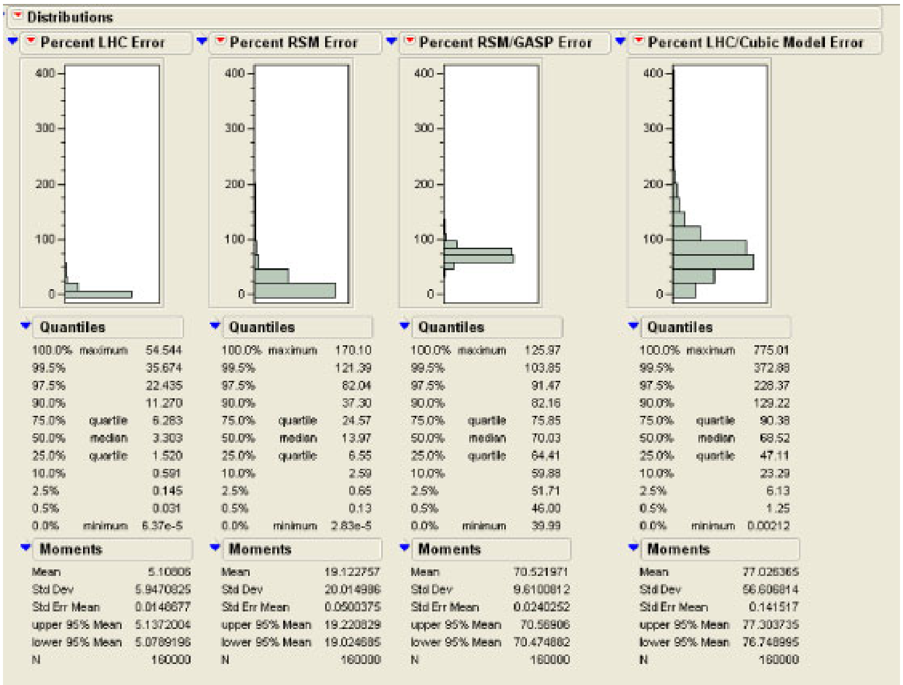
Figure 1 shows plots of two factor, 10 run designs for sphere packing, Latin Hypercube, Uniform, Maximum entropy, and GASP IMSE designs.

Currently, the most popular design for computer experiments is the LHD. We use this design for our example in the next section to illustrate the design and analysis of a computer experiment. As previously mentioned, in an LHD each column is a permutation of the numbers from 1 to n. Thus, there are potentially a huge number of LHDs for any given number of factors and runs. To make the design unique, a secondary criterion is usually specified. Examples are maximin LHDs that maximize the minimum distance between points, and orthogonal LHDs that minimize the column correlations. Note that using random LHDs may not be a good idea as seen by examining Figure 2.

4. EXAMPLE OF THE DESIGN AND ANALYSIS OF A COMPUTER EXPERIMENT

In the introduction, we gave several examples of computer experiments. They include FEA models, CFD models, and circuit simulation models. These models are often based on a large set of differential equations that are simultaneously evaluated. One way to investigate the properties of designs and analyses for computer simulations is through a known test function. This test function acts in place of a computer simulation model. The test function in this example is the F quantile function. The noncentral F cumulative distribution function (cdf) is

$$F(x|v_1, v_2, \delta) = \sum_{j=0}^{\infty} \left(\frac{\left(\frac{1}{2}\delta\right)^2}{j!} \mathrm{e}^{-\delta/2} \right) I\left(\frac{v_1 x}{v_2 + v_1 x} \middle| \frac{v_1}{2} + j, \frac{v_2}{2} \right)$$



Quantiles

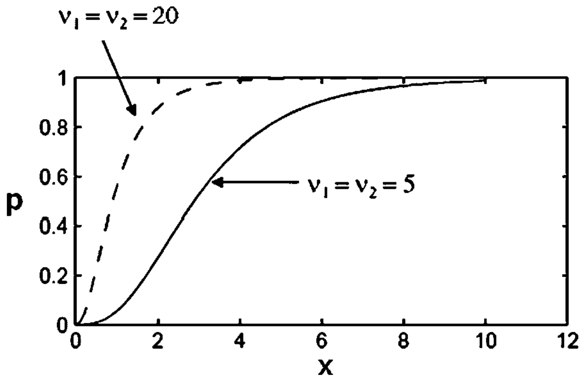
		Percent LHC Error	Percent RSM Error	Percent RSM/GASP Error	Percent LHC/Cubic Model Error
100.0%	maximum	54.544	170.10	125.97	775.01
99.5%		35.674	121.39	103.85	372.88
97.5%		22.435	82.04	91.47	228.37
90.0%		11.270	37.30	82.16	129.22
75.0%	quartile	6.283	24.57	75.85	90.38
50.0%	median	3.303	13.97	70.03	68.52
25.0%	quartile	1.520	6.55	64.41	47.11
10.0%		0.591	2.59	59.88	23.29
2.5%		0.145	0.65	51.71	6.13
0.5%		0.031	0.13	46.00	1.25
0.0%	minimum	6.37e-5	2.83e-5	39.99	0.00212

Moments

	Percent LHC Error	Percent RSM Error	Percent RSM/GASP Error	Percent LHC/Cubic Model Error
Mean	5.10806	19.122757	70.521971	77.026365
Std Dev	5.9470825	20.014986	9.6100812	56.606814
Std Err Mean	0.0148677	0.0500375	0.0240252	0.141517
upper 95% Mean	5.1372004	19.220829	70.56906	77.303735
lower 95% Mean	5.0789196	19.024685	70.474882	76.748995
N	160000	160000	160000	160000

Figure 3. Percent error summaries of each of the four design and analysis cases using the F quantile test function. This figure is available online at www.interscience.wiley.com/journal/qre

where $I(x, a|b)$ is the incomplete beta function with parameters a and b. The function can be found in Johnson and Kotz[39]. A graph of the function, with noncentrality parameter, $\delta = 10$, is shown in Figure 3. The noncentral F cdf provides the probability of observing a value more extreme than x, given ν_1, ν_2, and δ. The F quantile function is the inverse function of the F cdf. That is, it provides the critical value, x, given p, ν_1, ν_2, and δ.



The usefulness of any computer simulation experiment depends on both the design and the model that is fit. We consider two designs and two modeling approaches. The two designs are the maximin LHD and a D-optimal design for fitting the full cubic model. Both designs include the four factors, which are the parameters of the F quantile function (p, ν_1, ν_2, and δ) and both have 35 points. The two models are the GASP model and the cubic polynomial response surface model. Note that the cubic model has 35 parameters, hence the least-squares fit of the 35 point design will interpolate the data. The combinations of design and model lead to the four test cases below:

- *LHD-GASP*: Latin Hypercube design fit with the GASP model.
- *LHD-CP*: Latin Hypercube design fit with the cubic polynomial model.
- *RSM-GASP*: D-optimal design fit with the GASP model.
- *RSM-CP*: D-optimal design fit with the cubic polynomial model.

We evaluate the design of each case using percent error. To do this we first created a 20^4 point grid of parameter values for p, v_1, v_2, and δ. We used the prediction model derived from each of the four cases to predict the value of these 160 000 points in the design space and calculated the percent error. Note that since this is a known function, we can calculate the exact F-quantile for each of these 160 000 points.

Figure 3 displays the results graphically and numerically. The left-most histogram in Figure 3 (percent LHD error) shows the percent error for the LHD fit with the GASP model (LHD-GASP). Moving to the right, the next histogram in Figure 3, labeled Percent RSM error, shows the percent error generated by the D-optimal design fit with the cubic polynomial model (RSM-CP). The next histogram, labeled Percent RSM/GASP Error, shows the percent error for the D-optimal design fit with the GASP model (RSM-GASP). The right most histogram, labeled Percent LHD/Cubic Model Error, shows the percent error of the LHD fit with the cubic polynomial model (LHD-CP). From the histograms and the descriptive statistics below them, we conclude that the LHD combined with the GASP model far outperforms the other alternatives. Note, however, that there is a strong interaction involving the design and analysis strategy. The worst performing combination also involves the LHD but fit with the cubic regression model.

5. CONCLUSIONS

The design and analysis of computer experiments are research areas with growing impact. Our goal in this paper was to provide an accessible introduction to the area. An example demonstrates that traditional design and analysis techniques may not perform as well as the newer methods discussed. Our example employed a fairly well behaved test function. It is notable that an LHD fit with a GASP model performed much better than an optimal design for a low-order polynomial combined with a least-squares fit of that polynomial.

Finally, it is important to keep in mind the strong relationship between the design and the model used to fit the data. The design and the model should match. The LHD worked best when coupled with a GASP model. When coupled with a cubic regression model, it performed worst of all. Similarly, the D-optimal design for a cubic model worked better when the cubic model was fit. Results were less desirable when fitting a GASP model to a D-optimal design.

REFERENCES

1. Currin C, Mitchell TJ, Morris MD, Ylvisaker D. Bayesian prediction of deterministic functions, with applications to the design and analysis of computer experiments. *Journal of the American Statistical Association* 1991; **86**:953–963.
2. Currin C, Mitchell TJ, Morris MD, Ylvisaker D. A Bayesian approach to the design and analysis of computer experiments. *ORNL-6498*, Available from National Technical Information Service, 5285 Port Royal Road, Springfield, VA 22161, 1988.
3. Sacks J, Welch WJ, Mitchell TJ, Wynn HP. Design and analysis of computer experiments. *Statistical Science* 1989; **4**(4):409–423.
4. Allen TT, Bernshteyn MA, Kabiri-Bamoradian K. Constructing meta-models for computer experiments. *Journal of Quality Technology* 2003; **35**(3):264–274.
5. Johnson ME, Watson CC. Fitting statistical distributions to data in hurricane modeling. *American Journal of Mathematical and Management Sciences* 2006.
6. Iman RL, Johnson ME, Watson CC. Statistical aspects of forecast planning for hurricanes. *The American Statistician* 2006; **60**(2):1713–1726.

7. Watson CC, Johnson ME. Hurricane loss estimation models: Opportunities for improving the state of the art. *Bulletin of the American Meteorlogical Society* 2004; **85**:1713–1726.
8. Xiao X, Edwards JR, Hassan HA, Cutler A. Variable turbulent Schmidt-number formulation for scramjet applications. *AIAA Journal* 2006; **44**(3):593–599.
9. Park JS. Tuning complex computer codes to data and optimal designs. *PhD Thesis*, University of Illinois, Champaign/Urbabna, IL, 1991.
10. Craig PC, Goldstein M, Rougier JC, Seheult AH. Bayesian forecasting for complex systems using computer simulators. *Journal of the American Statistical Association* 2001; **96**:717–729.
11. Kennedy MC, O'Hagan A. Bayesian calibration of computer models (with discussion). *Journal of the Royal Statistical Society B* 2001; **63**:425–464.
12. Trucano TG, Pilch M, Oberkamph WL. General concepts for experimental validation of ASCII code applications. *Technical Report SAND2002-0341*, Sandia National Laboratories, 2002.
13. Pilch M, Trucano TG, Moya J, Froehlich G, Hodges A, Peercy D. Guidelines for Sandia ASCI verification and validation plans content and format: Version 2.0. *Technical Report SAND2000-3101*, Sandia National Laboratories, 2000.
14. Roache PJ. *Verification and Validation in Computational Science and Engineering*. Hermosa Publishers: Albuquerque, NM, 1998.
15. Hills RG, Trucano TG. Statistical validation of engineering and scientific models: Background. *Technical Report SAND99-1256*, Sandia National Laboratories, 1999.
16. Bayarri MJ, Berger JO, Paulo R, Sacks J, Cafeo JA, Cavendish J, Lin CH, Tu J. A framework for validation of computer models. *Technometrics* 2007; **49**(2):138–154.
17. Berk R, Bickel P, Campbell K, Fovell R, Keller-McNulty S, Kelly E, Linn R, Park B, Perelson A, Rouphail N, Sacks J, Schoenberg F. Workshop on statistical approaches for the evaluation of complex computer models. *Statistical Science* 2002; **17**:173–192.
18. Sacks J, Rouphail NM, Park B, Thakuriah P. Statistically-based validation of computer simulation models in traffic operations and management. *Technical Report 112*, National Institute of Statistical Sciences, 2000.
19. Santner TJ, Williams BJ, Notz WI. *The Design and Analysis of Computer Experiments*. Springer Series in Statistics. Springer: New York, 2003.
20. Johnson RT, Montgomery DC, Jones B, Parker PA. Comparing computer experiments using high order polynomial metamodels. *Journal of Quality Technology* 2009; submitted.
21. Matheron G. Principles of geostatistics. *Economic Geology* 1963; **58**:1246–1266.
22. Johnson ME, Moore LM, Ylvisaker D. Minimax and maxmin distance design. *Journal for Statistical Planning and Inference* 1990; **26**:131–148.
23. Jank W, Shmueli G. Modelling concurrency of events in on-line auctions via spatiotemporal semiparametric models. *Applied Statistics* 2007; **56**:1–27.
24. Liefvendahl M, Stocki R. A study on algorithms for optimization of Latin hypercubes. *Journal of Statistical Planning and Inference* 2006; **136**:3231–3247.
25. Chen V, Tsui KL, Barton R, Meckensheime M. A review on design, modeling and applications of computer experiments. *IEE Transactions* 2006; **38**:273–291.
26. Roux W, Stander N, Gunther F, Mullerschon H. Stochastic analysis of highly non-Linear structures. *International Journal for Numerical Methods in Engineering* 2006; **65**:1221–1242.
27. Bursztyn D, Steinberg DM. Comparison of designs for computer experiments. *Journal of Statistical Planning and Inference* 2006; **136**:1103–1119.
28. McKay ND, Conover WJ, Beckman RJ. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics* 1979; **21**:239–245.
29. Fang KT, Li R, Sudjianto A. *Design and Modeling for Computer Experiments*. Taylor & Francis Group: Boca Raton, 2006.
30. Welch WJ, Buck RJ, Sacks J, Wynn HP, Mitchell TJ, Morris MD. Screening, predicting, and computer experiments. *Technometrics* 1992; **34**(1):15–25.
31. Mease D, Bingham D. Latin hyperrectangle sampling for computer experiments. *Technometrics* 2006; **48**(4):467–477.
32. Tyre A, Kerr GD, Tenhumberg B, Bull M. Identifying mechanistic models of spatial behaviour using pattern-based modelling: An example from lizard home ranges. *Ecological Modeling* 2007; **208**:307–316.
33. Storlie CB, Helton JC. Multiple predictor smoothing methods for sensitivity analysis: Example results. *Reliability Engineering and System Safety* 2008; **93**:55–77.
34. Fang KT. The uniform design: Application of number–theoretic methods in experimental design. *Acta Mathematicae Applicatae Sinica* 1980; **3**:363–372.

DOI: 10.1002/qre

35. Wang Y, Fang KT. A note on uniform distribution and experimental design. *KeXue TongBao* 1981; **26**:485–489.
36. Hickernell FJ. A generalized discrepancy and quadrature error bound. *Mathematics of Computation* 1998; **67**:299–322.
37. Shewry MC, Wynn HP. Maximum entropy sampling. *Journal of Applied Statistics* 1987; **14**:898–914.
38. Ko CW, Lee J, Queyranne M. An exact algorithm for maximum entropy sampling. *Operations Research* 1995; **43**:684–691.
39. Johnson N, Kotz S. *Distributions in Statistics Continuous*: *Univariate Distributions-2*. Wiley: New York, 1970; 189–200.
40. Hussain MF, Barton RR, Joshi SB. Metamodeling: Radial basis functions, versus polynomials. *European Journal of Operational Research* 2002; **138**:142–154.

Authors' biographies

Bradley Jones is the Director of Research & Development in the JMP Division of SAS Institute. He received his MS in Statistics from Florida State University and his PhD in Applied Economic Sciences from the University of Antwerp. He is a Fellow of the American Statistical Association.

Rachel T. Johnson is an Assistant Professor in the Operations Research Department at the Naval Postgraduate School. She received her BS in Industrial Engineering from the Northwestern University and her MS and PhD in Industrial Engineering from Arizona State University.

Published in 2009 by John Wiley & Sons, Ltd.

QUALITY AND RELIABILITY ENGINEERING INTERNATIONAL
Qual. Reliab. Engng. Int. 2009; **25**:525–527
Published online 18 June 2009 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/qre.1038

Discussion

Discussion (1): Jones–Johnson Paper

Jones and Johnson (JJ) offer a clear and helpful introduction to ideas and methods currently used in the design and analysis of computer experiments, especially those based on Gaussian Stochastic Process (GaSP) models. I will briefly mention some additional points of importance regarding analysis (related to their presentation in Section 2), design (Section 3), and a final note on computing.

1. ANALYSIS

JJ describe prediction derived from a GaSP model, concluding Section 2 with the form of the point predictions and relative prediction variances based on 'plugging in' the maximum likelihood estimates of μ, σ, and θ's as if they were the actual parameter values. This approach is relatively simple, and often works quite well in practice. Much recent work is based on a full Bayesian approach, including the specification of priors for the GaSP parameters, and resulting in predictions that more fully account for parameter uncertainty. While the required computing is more intensive than with the 'plug-in' approach, it is generally quite feasible using modern Bayesian computing techniques, and software for accomplishing this is now widely available (e.g. GEM-SA, available at http://ctcd.group.shef.ac.uk/gem.html).

Many computer models produce outputs that are functional, rather than scalar-valued. For example, climate model output is indexed both by time and space, a common feature of models of dynamic systems. In some cases, it may be adequate to reduce the functional outputs to one or a few scalar-valued variables so that methodology similar to that described by JJ can be employed. However, these relatively few summary values must contain enough information to accurately reconstruct the functional output if the surrogate is to be used for prediction. Fortunately, the functional outputs of many models have similar overall structure from run to run, and techniques based on principal component representations can be very effective; see for example Higdon *et al.*[1].

Computer models often describe functions that are fairly well-behaved over most of the design region, but may be more erratic here and there. For example, if one input is temperature, the behavior of a physical system including water may be reasonably predictable above or below the freezing point, but the transition between the two regions may behave in a substantially different fashion. In other cases, models display the beginning of asymptotic behavior in some inputs; outputs may be quite sensitive to changes in small values of x, but become relatively stable toward the upper end of the input's range. When this occurs, *stationary* GaSP covariance structures such as the one described by JJ, and used in most applications of GaSP models to computer experiments, can sometimes lead to predictions of poor quality. However the prior specification of a more appropriate non-stationary form is usually impractical. Drignei and Morris[2] present one solution to this problem for computer models that evaluate systems of differential equations, in which a stationary GaSP is used to model intermediate results called *numerical truncation errors* that in turn are nonlinearly transformed to output predictions, effectively allowing the data to determine the form of an appropriate non-stationary process. Gramacy and Lee[3] develop a more general purpose approach in which the input space is partitioned and a (potentially) different stationary GaSP is used in each partition.

2. DESIGN

To augment JJ's description of the maximin (Mm) distance criterion and the entropy criterion, it is worth noting that these two are actually closely related, as discussed in Johnson, Moore, and Ylvisaker (1990, JMY). Briefly and in JJ's notation, the entropy criterion leads to the selection of a design that maximizes the determinant of the prior $n \times n$ correlation matrix at the design points, R (requiring that θ's be known or that values for them be stipulated for design purposes). The Mm distance design criterion, as defined by JMY, leads to the selection of a design for which the smallest Euclidean distance between two design points is maximized *and*, from among the designs that accomplish this, the number of pairs of design points separated by this distance is minimized. (Euclidean distance is tied to the correlation function used by JJ; other distance measures are associated with other correlation forms.) The theory presented by JMY shows that as the correlation over any fixed distance becomes weak (i.e. the θ's in JJ's notation approach infinity), the Mm-optimal design is also entropy-optimal in the limit. So, especially in cases where it is expected that all inputs have substantial influence on the output, the Mm criterion may be a good substitute for the entropy criterion, which requires θ values to be specified, and for which the computing required for construction of an optimal design can be relatively demanding.

As noted by JJ, sequential computer experiments are often employed to determine the input values that lead to maximized or minimized output. Design and analysis are necessarily combined in this context since interim analyses are used to guide the evolving experimental design. The sequential *expected improvement criterion* described by Jones *et al.*[4] is developed for this situation.

3. COMPUTING

The degree of the computational burden involved in estimating GaSP parameters, or finding an optimal design via the entropy criterion, is largely determined by the size of the matrix R $(n \times n)$. The 'inner loop' of these calculations involves the computing of an inverse and/or determinant of this dense and often numerically ill-conditioned matrix. As a result, it can be very difficult or impossible to perform unmodified GaSP-related calculations for data sets of more than a few hundred model runs. While this is usually not a major limitation for the computer models that are expensive to execute, it is often practical to generate far more runs when the model executes quickly, and large computer experiments can be required to adequately characterize functions with a large number of active inputs. Research leading to practical GaSP-like predictions in high dimension, based on relatively large data sets, will be a valuable contribution to computer experiments methodology.

Acknowledgements

This work was partially funded through grant #DMS-0502347 from the National Science Foundation.

REFERENCES

1. Higdon D, Gattiker JR, Williams BJ. Computer model calibration using high dimensional output. *Journal of the American Statistical Association* 2008; **103**:570–583.
2. Drignei D, Morris MD. Empirical Bayesian analysis for computer experiments involving finite-difference codes. *Journal of the American Statistical Association* 2006; **101**:1527–1536.
3. Gramacy RB, Lee HKH. Bayesian treed gaussian process models with an application to computer modeling. *Journal of the American Statistical Association* 2008; **103**:1119–1130.

DOI: 10.1002/qre

4. Jones DR, Schonlau M, Welch WJ. Efficient global optimization of expensive black-box functions. *Journal of Global Optimization* 1998; **13**:455–492.

MAX D. MORRIS
Department of Statistics
Department of Industrial and
Manufacturing Systems Engineering
Iowa State University
Ames, *IA*, *U.S.A.*
E-mail: mmorris@iastate.edu

DOI: 10.1002/qre

QUALITY AND RELIABILITY ENGINEERING INTERNATIONAL
Qual. Reliab. Engng. Int. 2009; **25**:529–533
Published online 18 June 2009 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/qre.1039

Discussion

Discussion (2): Jones–Johnson Paper

We congratulate Jones and Johnson for their interesting article. They have presented a useful survey of the design and analysis of Gaussian process (GP) models for computer experiments, along with a case study. As commonly done in the literature, this article assumes that all the factors involved in a computer experiment are quantitative. However, computer modeling can also include qualitative factors. Consider, for example, the data-center computer experiment described by Schmidt *et al.*[1] The configuration variables that determine the thermal properties of a data center can be either quantitative or qualitative. Examples of quantitative variables are rack temperature rise, rack heat load, and total diffuser flow rate. Examples of qualitative variables are diffuser location, return air vent location, and rack heat load non-uniformity. To complement this article, we report some new GP models with qualitative and quantitative factors, followed by a brief discussion of the related design issue.

In Qian *et al.*[2], a general approach is proposed to build a *single* GP model across different values of qualitative and quantitative factors so as to borrow strength from all the observations. Suppose a computer experiment involves factors $\mathbf{w}=(\mathbf{x}^t,\mathbf{z}^t)^t$, where the factors in $\mathbf{x}=(x_1,\ldots,x_I)^t$ are quantitative, and the factors in $\mathbf{z}=(z_1,\ldots,z_J)^t$ are assumed to be categorical, but not ordinal. The response $y(\mathbf{w})$ at the input value $\mathbf{w}$ is assumed to be

$$y(\mathbf{w})=\boldsymbol{\beta}^t\mathbf{f}(\mathbf{w})+\varepsilon(\mathbf{w}) \tag{1}$$

where $\mathbf{f}(\mathbf{w})=(f_1(\mathbf{w}),\ldots,f_l(\mathbf{w}))^t$ is the vector of l pre-specified functions (e.g. polynomials) and $\boldsymbol{\beta}=(\beta_1,\ldots,\beta_l)^t$ is the vector of unknown coefficients. The residual $\varepsilon(\mathbf{w})$ is assumed to be a GP with mean zero, variance σ^2, and some correlation function. Construction of a 'valid' correlation function for $\varepsilon(\mathbf{w})$ is not straightforward because such a function needs to be defined in the space involving both qualitative and quantitative factors and the notion of 'distance' is absent for categorical factors.

Now we discuss the construction of correlation functions for $\varepsilon(\mathbf{w})$. First, consider the case with one qualitative factor, z_1, with m_1 levels, denoted by $1,\ldots,m_1$. To define the correlation function of $\varepsilon(\mathbf{w})$, where $\mathbf{w}=(\mathbf{x}^t,z_1)^t$, let $\varepsilon_u(\mathbf{x})=\varepsilon((\mathbf{x}^t,u)^t)$, for $u=1,\ldots,m_1$, and envision a mean-zero m_1-variate process $\boldsymbol{\varepsilon}^*(\mathbf{x})=(\varepsilon_1(\mathbf{x})\ldots\varepsilon_{m_1}(\mathbf{x}))^t$. Then we only need to define correlation and cross-correlation functions for $\boldsymbol{\varepsilon}^*(\mathbf{x})$. A convenient approach is to assume that $\boldsymbol{\varepsilon}^*(\mathbf{x})=\mathbf{A}\boldsymbol{\eta}(\mathbf{x})$, where $\mathbf{A}=(\mathbf{a}_1,\ldots,\mathbf{a}_{m_1})^t$ is an $m_1\times m_1$ non-singular matrix with unit row vectors (i.e. $\mathbf{a}_u^t\mathbf{a}_u=1$ for $u=1,\ldots,m_1$), and $\boldsymbol{\eta}(\mathbf{x})=(\eta_1(\mathbf{x}),\ldots,\eta_{m_1}(\mathbf{x}))^t$, where $\eta_1(\mathbf{x}),\ldots,\eta_{m_1}(\mathbf{x})$ are independent stochastic processes with the same variance σ^2 and correlation function $K_{\boldsymbol{\phi}}$. Then for input values $\mathbf{w}_i=(\mathbf{x}_i^t,z_{1i})^t$ $(i=1,2)$, the correlation function is

$$\operatorname{cor}(\varepsilon(\mathbf{w}_1),\varepsilon(\mathbf{w}_2))=\mathbf{a}_{z_{11}}^t\mathbf{a}_{z_{12}}K_{\boldsymbol{\phi}}(\mathbf{x}_1,\mathbf{x}_2) \tag{2}$$

Let $\tau_{r,s}=\mathbf{a}_r^t\mathbf{a}_s$, where $r,s=1,\ldots,m_1$. Then $\mathbf{T}_1=(\tau_{r,s})=\mathbf{A}\mathbf{A}^t$ is an $m_1\times m_1$ *positive-definite matrix with unit diagonal elements* (abbreviated to PDUDE). Next consider the general case with J qualitative factors $\mathbf{z}=(z_1,\ldots,z_J)^t$, where z_j has m_j levels, denoted by $1,\ldots,m_j$, for $j=1,\ldots,J$. Here a correlation function for $\varepsilon(\mathbf{w})$ can be constructed as

$$\operatorname{cor}(\varepsilon(\mathbf{w}_1),\varepsilon(\mathbf{w}_2))=\prod_{j=1}^{J}[\tau_{j,z_{j1},z_{j2}}K_{\boldsymbol{\phi}_j}(\mathbf{x}_1,\mathbf{x}_2)] \tag{3}$$

Table I. The matrix B

Run #	x_1	x_2	x_3	x_4	x_5
1	1	1	1	1	1
2	1	3	3	5	7
3	1	5	5	8	4
4	1	7	7	4	6
5	1	8	8	7	2
6	1	6	6	3	8
7	1	4	4	2	3
8	1	2	2	6	5
9	3	1	3	3	3
10	3	3	1	7	5
11	3	5	7	6	2
12	3	7	5	2	8
13	3	8	6	5	4
14	3	6	8	1	6
15	3	4	2	4	1
16	3	2	4	8	7
17	5	1	5	5	5
18	5	3	7	1	3
19	5	5	1	4	8
20	5	7	3	8	2
21	5	8	4	3	6
22	5	6	2	7	4
23	5	4	8	6	7
24	5	2	6	2	1
25	7	1	7	7	7
26	7	3	5	3	1
27	7	5	3	2	6
28	7	7	1	6	4
29	7	8	2	1	8
30	7	6	4	5	2
31	7	4	6	8	5
32	7	2	8	4	3
33	8	1	8	8	8
34	8	3	6	4	2
35	8	5	4	1	5
36	8	7	2	5	3
37	8	8	1	2	7
38	8	6	3	6	1
39	8	4	5	7	6
40	8	2	7	3	4
41	6	1	6	6	6
42	6	3	8	2	4
43	6	5	2	3	7
44	6	7	4	7	1
45	6	8	3	4	5
46	6	6	1	8	3
47	6	4	7	5	8
48	6	2	5	1	2
49	4	1	4	4	4
50	4	3	2	8	6
51	4	5	8	5	1
52	4	7	6	1	7
53	4	8	5	6	3
54	4	6	7	2	5
55	4	4	1	3	2
56	4	2	3	7	8
57	2	1	2	2	2

DOI: 10.1002/qre

Table I. *Continued*

Run #	x_1	x_2	x_3	x_4	x_5
58	2	3	4	6	8
59	2	5	6	7	3
60	2	7	8	3	5
61	2	8	7	8	1
62	2	6	5	4	7
63	2	4	3	1	4
64	2	2	1	5	6

where $\mathbf{T}_j=(\tau_{j,r,s})$ is an $m_j\times m_j$ PDUDE. One popular choice of $K_{\boldsymbol{\phi}_j}(\mathbf{x}_1,\mathbf{x}_2)$ is $\exp\{-\sum_{i=1}^{I}\phi_{ij}(x_{i1}-x_{i2})^p\}$.

Next we discuss the estimation of the foregoing models. Suppose the data consist of n different input values, $\mathbf{D}_w=(\mathbf{w}_1^0,\ldots,\mathbf{w}_n^0)^{\mathrm{t}}$, and the corresponding responses, $\mathbf{y}=(y_1,\ldots,y_n)^{\mathrm{t}}$. The parameters to be estimated are $\boldsymbol{\beta}=(\beta_1,\ldots,\beta_l)^{\mathrm{t}}$, σ^2, $\boldsymbol{\phi}=(\phi_1,\ldots,\phi_I)^{\mathrm{t}}$, and $\mathbf{T}=\{\mathbf{T}_1,\ldots,\mathbf{T}_J\}$. We use the method of maximum likelihood for estimation, and denote the resulting estimators by $\widehat{\boldsymbol{\beta}}$, $\widehat{\sigma}^2$, $\widehat{\boldsymbol{\phi}}$, and $\widehat{\mathbf{T}}$. The log-likelihood of $\mathbf{y}$ is proportional to

$$(-\tfrac{1}{2})[n\ln\sigma^2+\ln|\mathbf{R}|+(\mathbf{y}-\mathbf{F}\boldsymbol{\beta})^{\mathrm{t}}\mathbf{R}^{-1}(\mathbf{y}-\mathbf{F}\boldsymbol{\beta})/\sigma^2] \tag{4}$$

where $\mathbf{F}=(\mathbf{f}(\mathbf{w}_1^0),\ldots,\mathbf{f}(\mathbf{w}_n^0))^{\mathrm{t}}$ is an $n\times l$ matrix; $\mathbf{R}$ is the correlation matrix, which depends on the correlation parameters $\boldsymbol{\phi}$ and $\mathbf{T}$, and its (i,j)th entry is $\mathrm{cor}(\varepsilon(\mathbf{w}_i^0),\varepsilon(\mathbf{w}_j^0))$. The estimates can be obtained by iterating between two steps:

Regression fitting: Given $\widehat{\boldsymbol{\phi}}$ and $\widehat{\mathbf{T}}$, $\widehat{\boldsymbol{\beta}}$ and $\widehat{\sigma}^2$ are obtained as follows:

$$\widehat{\boldsymbol{\beta}}=(\mathbf{F}^{\mathrm{t}}[\mathbf{R}(\widehat{\boldsymbol{\phi}},\widehat{\mathbf{T}})]^{-1}\mathbf{F})^{-1}\mathbf{F}^{\mathrm{t}}[\mathbf{R}(\widehat{\boldsymbol{\phi}},\widehat{\mathbf{T}})]^{-1}\mathbf{y} \quad \text{and} \quad \widehat{\sigma}^2=(1/n)(\mathbf{y}-\mathbf{F}\widehat{\boldsymbol{\beta}})^{\mathrm{t}}\mathbf{R}^{-1}(\mathbf{y}-\mathbf{F}\widehat{\boldsymbol{\beta}})$$

Correlation fitting: Given $\widehat{\boldsymbol{\beta}}$ and $\widehat{\sigma}^2$, let $u_i=(y_i-\widehat{\boldsymbol{\beta}}^{\mathrm{t}}\mathbf{f}(\mathbf{w}_i))/\widehat{\sigma}$, for $i=1,\ldots,n$. Then fit the proposed GP model with mean 0, variance 1, and correlation matrix $\mathbf{R}$ to the data $\mathbf{u}=(u_1,\ldots,u_n)^{\mathrm{t}}$, and estimate $\boldsymbol{\phi}$ and $\mathbf{T}$ using semi-definite programming techniques (Wolkowicz *et al.*[3]).

The empirical best linear unbiased predictor (BLUP) of y at the point $\mathbf{w}_0$ is

$$\widehat{y}(\mathbf{w}_0)=\widehat{\boldsymbol{\beta}}^{\mathrm{t}}\mathbf{f}(\mathbf{w}_0)+\widehat{\mathbf{r}}_0^{\mathrm{t}}\widehat{\mathbf{R}}^{-1}(\mathbf{y}-\mathbf{F}\widehat{\boldsymbol{\beta}}) \tag{5}$$

where $\widehat{\mathbf{r}}_0=(\widehat{\mathrm{cor}}(y(\mathbf{w}_0),y(\mathbf{w}_1^0)),\ldots,\widehat{\mathrm{cor}}(y(\mathbf{w}_0),y(\mathbf{w}_n^0)))^{\mathrm{t}}$ and this predictor smoothly interpolates all the observed data points.

Now we report a new type of design, called *sliced space-filling designs* (SSFDs), that accommodate both qualitative and quantitative factors, taken from Qian and Wu[4]. First we define a special type of orthogonal arrays (OAs). The definition of OAs can be found in Wu and Hamada[5]. Let B be an $OA(N_1,k,s_1,t)$. Suppose that the N_1 rows of this array can be partitioned into v subarrays each with N_2 rows, denoted by B_i, and the s_1 levels of B can be collapsed into s_2 levels with $s_1>s_2$ according to a rule δ. Further, suppose that B_i is an $OA(N_2,k,s_2,t)$ if the s_1 levels of B are collapsed according to δ. Then B is a sliced orthogonal array.

Let B be a sliced orthogonal array defined above. Construction of an SSFD is as follows. The array B is used to generate an OA-based Latin hypercube design D[6] for the quantitative factors, where the points corresponding to B_i are denoted by D_i. Then D_i's are associated with different level combinations of the qualitative factors. The array D is an SSFD. The proposed designs are intuitively appealing. They possess good space-filling properties when collapsed over the qualitative factors. Also, for any qualitative factor-level combination, the design points for the quantitative factors achieve uniformity in low dimensions.

For illustration, consider an example with five quantitative factors x_1 to x_5 and two qualitative factors z_1 and z_2 at two levels $-$ and $+$. Table I presents a sliced orthogonal array B, where B_{11}, B_{12}, B_{21}, B_{22} correspond to runs 1–4, 9–12, 17–20, 25–28; runs 57–60, 49–52, 41–44, 33–36; 5–8, 13–16, 21–24, 29–32; and runs

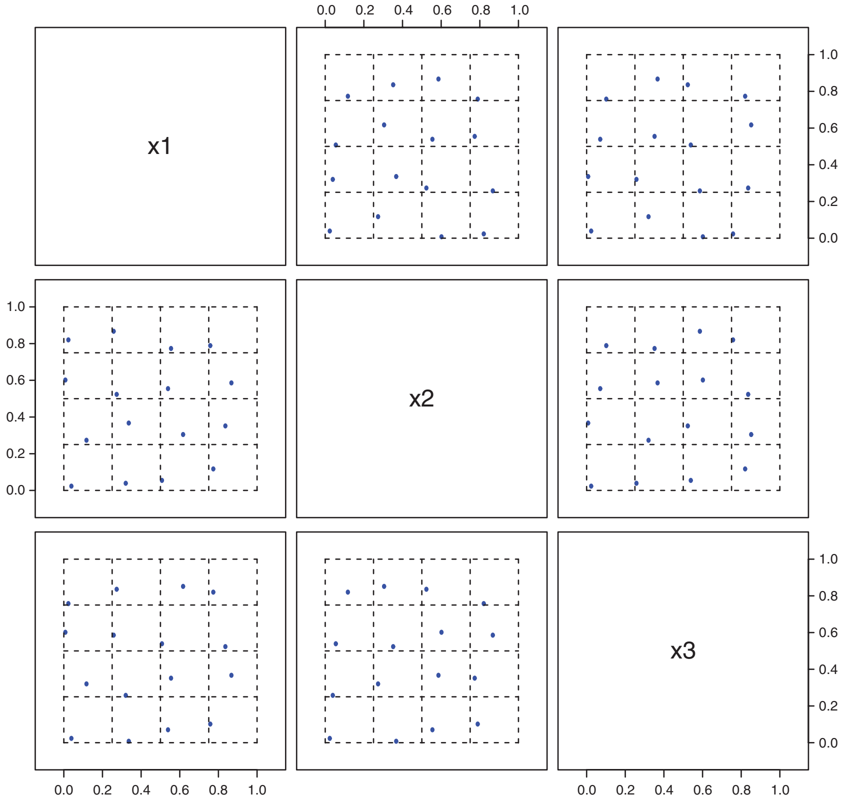


Figure 1. Bivariate projections among x_1, x_2, x_3 of D_{11}. This figure is available in color online at www.interscience.wiley.com/journal/qre

61–64, 53–56, 45–48, 37–40, respectively. We use B to construct an OA-based Latin hypercube design D for x_1 to x_5. Now partition D into D_{ij} with points corresponding to B_{ij}, where i and j are 1 and 2. Finally D_{11}, D_{12}, D_{21}, D_{22} are associated with the level combinations $(z_1, z_2) = (-, -)$, $(z_1, z_2) = (-, +)$, $(z_1, z_2) = (+, -)$, and $(z_2, z_2) = (+, +)$, respectively. For illustration, the bivariate projections of D_{11} are given in Figure 1, where the points achieve uniformity on 4×4 grids in two dimensions.

Acknowledgements

Qian's research was supported by NSF grant DMS-0705206 and a faculty award from IBM and Wu's research was supported by NSF grant DMS-0705261.

REFERENCES

1. Schmidt RR, Cruz EE, Iyengar MK. Challenges of data center thermal management. *IBM Journal of Research and Development* 2005; **49**:709–723.
2. Qian PZG, Wu H, Wu CFJ. Gaussian process models for computer experiments with qualitative and quantitative factors. *Technometrics* 2008; **50**:383–396.
3. Wolkowicz H, Saigal R, Vandenberghe L (eds.). *Handbook of Semidefinite Programming*: *Theory*, *Algorithms*, *and Applications*. Kluwer Academic: Boston, 2000.

DOI: 10.1002/qre

4. Qian PZG, Wu CFJ. Sliced space-filling designs. *Biometrika* 2009; to appear.
5. Wu CFJ, Hamada M. *Experiments*: *Planning*, *Analysis*, *and Parameter Design Optimization*. Wiley: New York, 2000.
6. Tang B. Orthogonal array-based Latin hypercubes. *Journal of the American Statistical Association* 1993; **88**:1392–1397.

PETER Z. G. QIAN
Department of Statistics
University of Wisconsin–Madison
Madison, *WI 53706*, *U.S.A.*
E-mail: peterq@stat.wisc.edu

C. F. JEFF WU
School of Industrial and Systems Engineering
Georgia Institute of Technology
Atlanta, *GA 30332*, *U.S.A.*
E-mail: jeffwu@isye.gatech.edu

DOI: 10.1002/qre

QUALITY AND RELIABILITY ENGINEERING INTERNATIONAL
Qual. Reliab. Engng. Int. 2009; **25**:535–539
Published online 18 June 2009 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/qre.1040

Discussion

Discussion (3): Jones–Johnson Paper

We commend Jones and Johnson for providing a clear and concise introduction to the quickly expanding field of statistical design and analysis of computer experiments. Our own experiences confirm that computer models are widely used in many areas of science and engineering, and the need to understand the performance of these models is critical. This is particularly true in light of the fact that with improving model fidelity, they are increasingly used to certify complex engineering systems with decreasing reliance on expensive physical experiments.

As the authors indicate, many computer models are sufficiently complex that only a small budget of model runs is allowed for any given application. Therefore, concepts of statistical experiment design become relevant for the purpose of intelligently selecting runs to inform the development of a statistical surrogate for model output—often referred to as an *emulator*—that will serve as the basis for statistical inference. There are several practical issues with emulating any computer model. Is the standard Gaussian Process (GP) model a good choice for general applications? What mean and covariance structure should one choose? How should the budget of runs be expended? The authors addressed these issues effectively in their article. We provide some additional perspective in what follows.

Extensive literature (see Sacks *et al.*[1], Santner *et al.*[2]) and experience suggest that the GP model is an ideal candidate for building an emulator. Ben-Ari and Steinberg[3] conducted an extensive simulation study comparing the GP model with a large class of competing models and found that GP-based emulation performs well in many situations. There are several practical considerations that must be addressed when using the GP model. Arguably the most important is how many runs are needed to adequately emulate the computer model. Loeppky *et al.*[4] argue that the often quoted rule of '$n=10d$' (i.e. 10 model runs per dimension) generally provides sufficient information for emulation. The design chosen for the example of this article comes close to attaining this target, at $8.75d$. In addition to run size considerations, it is important to calculate diagnostics that directly assess the quality of model fit. The root mean square error (RMSE) is an obvious criterion; however, holdout samples are often unavailable in practice. In such cases the cross-validated (CV)-RMSE (see Welch *et al.*[5]) and the individual CV residuals are useful.

The remainder of this discussion is structured to draw attention to additional connections between traditional response surface methodology (RSM) and analysis of computer experiments using the GP model. In particular, we focus on two basic components of response surface methods (see Box and Wilson[6]) for which recent developments have made analogues available for analysis of computer experiments: sensitivity analysis and sequential optimization. Sensitivity analysis refers to measuring the impact of input variations on output uncertainty. In particular, output uncertainty can be decomposed into main and interaction effects analogous to traditional analysis of variance (ANOVA), and sensitivity indices measuring the contribution of these individual effects to the total output variance can be computed (see Saltelli *et al.*[7], Oakley and O'Hagan[8], Schonlau and Welch[9]). Sequential optimization of computer models based on the expected improvement criteria has proven efficient and effective (see Jones *et al.*[10]). These optimization algorithms are global in the sense that they explore regions of the input space in which prediction is poor (potential for optima), while focusing in on regions of space containing optima with high probability.

Goodness-of-fit diagnostics, sensitivity analysis and sequential optimization are explored with two analyses of the example using the F-quantile function presented in this article. The first analysis (referred to as

ML for *Maximum Likelihood*) adopts a constant mean and a power exponential covariance structure with smoothness parameters fixed at 2.0 (infinitely differentiable realizations). All parameters (mean, variance, correlation range) are fit using maximum likelihood (see Sacks *et al.*[1]). The second analysis (referred to as FB for *Full Bayes*) adopts a zero mean (calculations are centered) and a power exponential covariance structure with smoothness parameters fixed at 2.0. All parameters (variance, range) are sampled from their joint posterior distribution based on a full Bayes analysis (see Higdon *et al.*[11], Williams *et al.*[12]).

The CV-RMSE from the ML analysis is small (0.59); however, a plot of the CV residuals shows one large value. This may indicate a problem with goodness-of-fit. This example has the benefit of holdout calculations for assessing out of sample prediction quality. The RMSE is small at 1.44 (relative to a range of 62); however, further inspection shows that the maximum absolute error is 43. These values are consistent with those obtained from the FB analysis, where the mean RMSE is 1.59 with a 99% credible interval of (1.41, 2.23) based on 1000 posterior samples. The mean maximum absolute error is 46 with a 99% credible interval of (43, 55). This presents additional cause for concern regarding emulator fidelity based on the 35-run design.

Figure 1 shows the main effect functions for the four inputs based on the FB analysis. In the FB analysis, the main effect variation in probability p accounts for 65.3% of the total variation, while the denominator degrees of freedom υ_2 accounts for an additional 9.5%. Table I presents the main, interaction and total effect sensitivity indices based on the FB analysis. The total effect of an input is defined to be the amount of output variance attributable to all effects involving that input. In particular, note the strong two-factor interaction (21.1%) between υ_2 and p. ANOVA results based on a full-factorial design are presented, showing that the 35-run design with FB analysis is unable to correctly partition variance between the main effect of p and the interaction effect between p and υ_2, although the variance summed over these two effects is approximately

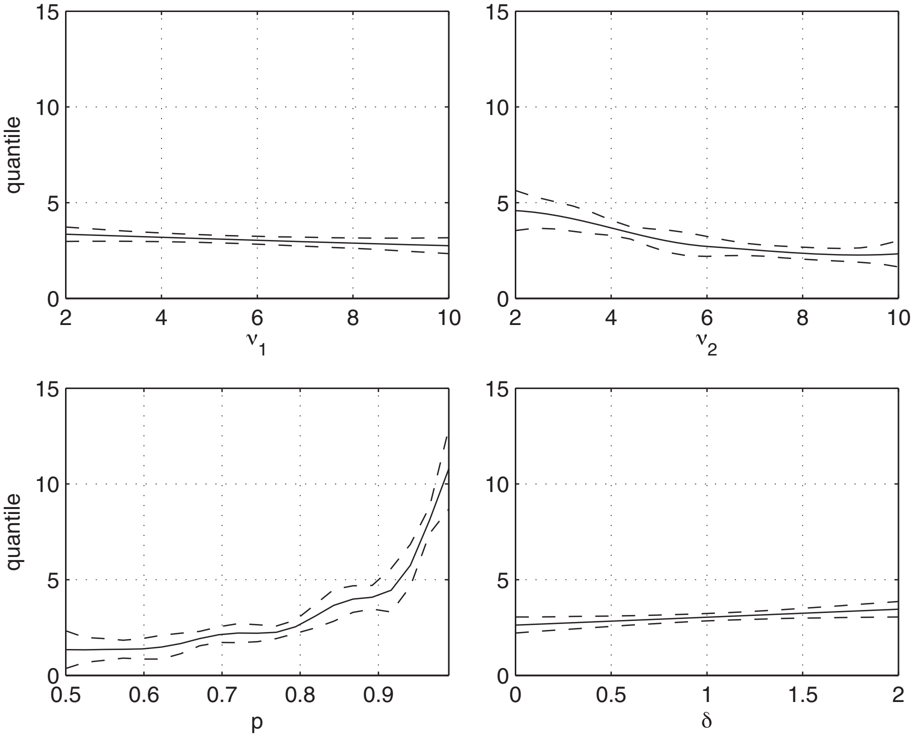
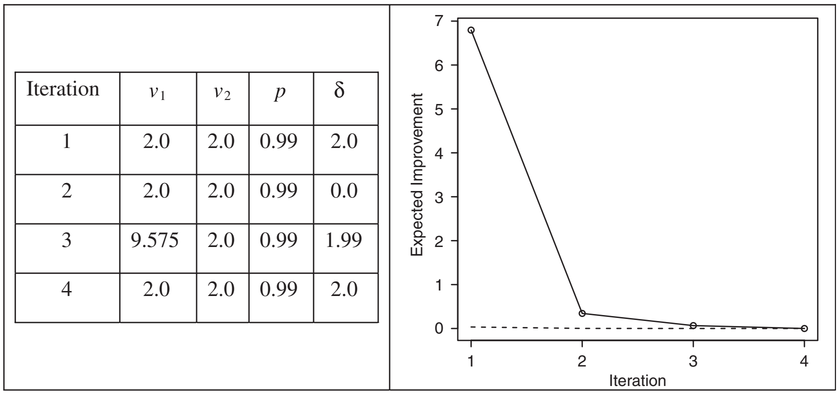


Figure 1. Main effect functions for the four inputs to the F-quantile function based on the FB analysis. The solid and dashed lines are the posterior mean and pointwise ±2 standard deviation bounds, respectively

DOI: 10.1002/qre

Table I. Main (on-diagonal) and interaction (off-diagonal) effects for the four inputs to the F-quantile function based on the FB analysis. Total effects for each input are presented in the last column. ANOVA results are presented in boldface

	v_1 (%)	v_2 (%)	p (%)	δ (%)	Total effect (%)
v_1 (%)	0.60 **0.31**	0.15 **0.04**	1.15 **0.67**	0.0 **0.08**	2.24
v_2 (%)		9.46 **9.49**	21.08 **54.54**	0.09 **0.10**	31.31
p (%)			65.26 **32.88**	0.63 **0.34**	88.65
δ (%)				1.04 **0.28**	1.98



Iteration	v_1	v_2	p	δ
1	2.0	2.0	0.99	2.0
2	2.0	2.0	0.99	0.0
3	9.575	2.0	0.99	1.99
4	2.0	2.0	0.99	2.0

Figure 2. Points added by the EGO algorithm (left panel); EI absolute (solid) and relative (dashed) criteria as a function of iteration (right panel)

the same in both analyses. This confirms the inadequacy of the 35-run design for emulation in at least a subspace of the input domain. In practice, it will not generally be possible to perform this ANOVA check of the emulator-based sensitivity analysis due to the large number of direct model runs required. However, as in this example, the observation of significant differences between ML- and FB-based sensitivity analyses provides an indication of emulator inadequacy.

In the ML analysis, the maximum error occurs at $(v_1, v_2, p, \delta) = (2.2, 2.2, 0.97775, 1.95)$. In fact, the 100 largest absolute errors occur for $(v_2, p) = (2.2, 0.97775)$, while v_1 and δ range from 2.2 to 7.4 and 0.65 to 1.95, respectively, consistent with results from the sensitivity analysis that v_2 and p explain most of the variation in the F-quantile function, while v_1 and δ are minor contributors. We note that the complexity in the F-quantile function for 'large p' and 'small v_2' (the F-distribution does not even have a variance for $v_2 \leq 4$) would be more effectively emulated with a design strategy that sequentially adds points to an initial set of runs for improving global fit (see Lam *et al.*[13]).

In addition to visualization one can also efficiently optimize the computer model sequentially. Figure 2 shows the results of applying the Efficient Global Optimization (EGO) algorithm of Jones *et al.*[10], using the Stochastic Process Analysis of Computer Experiments (SPACE) software provided by Dr. Matthias Schonlau (http://www.schonlau.net/space.html), to maximize the F-quantile function over the assumed input domain. The left panel gives the four points added by this algorithm, while the right panel shows the progression of the expected improvement (EI) criteria. The values of v_2 and p added by the EGO algorithm at each step

DOI: 10.1002/qre

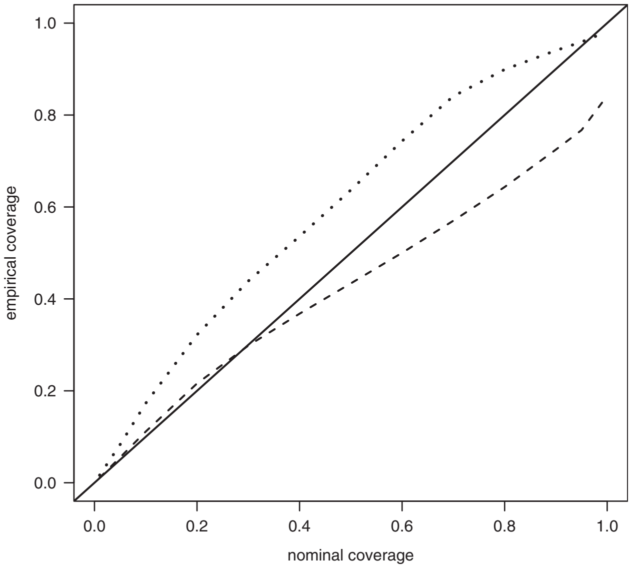


Figure 3. Plot of empirical coverage vs nominal coverage for the ML plug-in predictor (dashed) and the FB predictor (dotted)

are identical (2.0 and 0.99, respectively). The first point added is actually the optimum, selected because it is a good candidate for maximizing the output based on having large predicted response as well as large prediction uncertainty. The next two iterates explore changes to δ and v_1, and the last iteration returns to the vicinity of the optimum with values of both the absolute and relative EI criteria small enough to terminate the algorithm.

One of the major concerns with the GP is estimation of the parameters. The authors discuss the ML estimator, and indeed this estimator is frequently applied in practice. Furthermore, the ML estimator is often 'plugged-in' to the kriging prediction and variance formulas for the purpose of computer model emulation at unsampled inputs; however, it is well known that plug-in predictors tend to drastically underestimate the prediction error. As an alternative, an FB analysis generally provides superior coverage as parameter uncertainty is accounted for. Figure 3 illustrates these concepts on the 160 000 out of sample calculations provided by the authors. In this example, the ML plug-in analysis does not achieve nominal coverage at higher levels, while the FB analysis tends to be conservative although it is close to nominal coverage for levels often chosen in practice (e.g. 90, 95, 99%). As an alternative to FB, Nagy *et al.*[14] proposed an approximate Bayesian solution that is based on the ML parameter estimates and avoids some of the potential problems in properly tuning a Markov chain Monte Carlo sampler. Simulation studies show that this approximate method attains a remarkable match between actual and nominal coverage across the complete spectrum of levels.

We conclude by affirming the power of the GP model for emulation of computer models as pointed out by the authors. However, as highlighted by the analysis of this discussion, it is necessary to carefully examine multiple diagnostics for the purpose of assessing quality-of-fit. Cross validation is a useful technique in this regard, and out of sample validation is desirable if additional model runs can be made available. Techniques for conducting sensitivity analysis and sequential design for global prediction and optimization—analyses fundamental to traditional RSM—have been developed for computer experiments. Therefore, the flexibility of the RSM toolkit is available to practitioners who desire to take full advantage of GP emulation for computer experiments. Finally, we note the advances in prediction methodology for GP emulation that provides prediction uncertainties with good frequentist coverage properties, desirable in particular for external reviews of results based on GP modeling.

DOI: 10.1002/qre

REFERENCES

1. Sacks J, Welch WJ, Mitchell TJ, Wynn WP. Design and analysis of computer experiments. *Statistical Science* 1989; **4**:409–423.
2. Santner TJ, Williams BJ, Notz WI. *The Design and Analysis of Computer Experiments*. Springer: New York, 2003.
3. Ben-Ari EN, Steinberg DM. Modeling data from computer experiments: An empirical comparison of kriging with MARS and projection pursuit regression. *Quality Engineering* 2007; **19**:327–338.
4. Loeppky JL, Sacks J, Welch WJ. Choosing the sample size of a computer experiment: A practical guide. *Technical Report No. 170*, National Institute for Statistical Sciences, 2008. Available at: http://www.niss.org/downloadabletechreports.html [2008].
5. Welch WJ, Buck RJ, Sacks J, Wynn HP, Mitchell TJ, Morris MD. Screening, predicting, and computer experiments. *Technometrics* 1992; **34**(1):15–25.
6. Box GEP, Wilson KB. On the experimental attainment of optimum conditions. *Journal of the Royal Statistical Society, Series B* 1951; **13**:1–45.
7. Saltelli A, Chan K, Scott E. *Sensitivity Analysis*. Wiley: Chichester, 2000.
8. Oakley JE, O'Hagan A. Probabilistic sensitivity analysis of complex models: A Bayesian approach. *Journal of Royal Statistical Society B* 2004; **66**(3):751–769.
9. Schonlau M, Welch WJ. Screening the input variables to a computer code via analysis of variance and visualization. *Screening: Methods for Experimentation in Industry, Drug Discovery and Genetics*, AM Dean, SM Lewis (eds.). Springer: New York, 2006; 308–327.
10. Jones DR, Schonlau M, Welch WJ. Efficient global optimization of expensive black-box functions. *Journal of Global Optimization* 1998; **13**:455–492.
11. Higdon D, Kennedy M, Cavendish JC, Cafeo JA, Ryne RD. Combining field data and computer experiments for calibration and prediction. *SIAM Journal on Scientific Computing* 2004; **26**(2):448–466.
12. Williams B, Higdon D, Gattiker J, Moore L, McKay M, Keller-McNulty S. Combining experimental data and computer simulations, with an application to flyer plate experiments. *Bayesian Analysis* 2006; **1**(4):765–792.
13. Lam CQ, Notz WI. Sequential adaptive designs in computer experiments for response surface model fit. *Statistics and Applications* 2008; **6**(1–2):191–212.
14. Nagy B, Loeppky JL, Welch WJ. Fast Bayesian implementation of Gaussian process models. *UBC Technical Report No. 230*, 2007.

JASON L. LOEPPKY
University of British Columbia Okanagan
Kelowna, BC, Canada
E-mail: jason@stat.ubc.ca

BRIAN J. WILLIAMS
Los Alamos National Laboratory
Los Alamos, NM, U.S.A.
E-mail: brian@lanl.gov

DOI: 10.1002/qre

QUALITY AND RELIABILITY ENGINEERING INTERNATIONAL
Qual. Reliab. Engng. Int. 2009; **25**:541–543
Published online 18 June 2009 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/qre.1041

Discussion

Discussion (4): Jones–Johnson Paper

Jones and Johnson describe the use of computer simulators in experimentation, with emphasis on issues and criteria for good design and on the Gaussian Process (GASP) approach to modeling the data. They have done an excellent job of providing a readable and convincing introduction.

Computer simulation has become the experimental test bench in many scientific and engineering projects. Kenett and Steinberg[1] described a number of applications—design of an aircraft to improve structural strength, crash-safety tests for automobiles, and achieving a robust high-pressure compressor for an aeronautic engine. Thomke[2] provided an interesting business perspective on how computer experiments can accelerate the development process. Another important area of application is sensitivity analysis[3]; I will discuss an example later.

On the design side, Jones and Johnson discuss the implications of some special features that are common in computer experiments. With a deterministic simulator, repeat runs at the same input settings produce identical output and replication is foolish. Of course, not all simulators are deterministic. For example, simulators of queueing systems invariably have stochastic components representing the arrival and service times of jobs or customers. Results of these simulators thus include random errors (though often not with equal variance) and here standard design methods, including replication, *are* useful[4].

The experimental plans cited by Jones and Johnson use many levels for each factor; in the popular Latin hypercube (LHC) design, the number of levels is equal to the size of the sample. Note the sharp contrast to classical factorial designs, where typically just a few levels are used. The wisdom here is that the design should look more or less uniform, not just at the multi-factor level, but also if it is projected onto just a few 'active factors'; unique levels can help to achieve this goal. There is also a logistic aspect, with a tacit assumption that the levels of a simulator input, unlike those in a physical experiment, can be set on an arbitrarily fine grid, with no cost or inconvenience. This is not always the case, however. I was recently involved in a project where two of the factors described the geometry of the region under study and new levels for either of those factors required preparation of a new finite element scheme for the simulator. We used three levels for one factor and four for the other, thus needing 12 finite element grids. In another application, on the design of an airplane wing, each experimental setting had to be translated, manually, into a CAD diagram of the wing for use with the simulator. Here the size of the computer experiment was limited by the engineering time needed for the CAD layout of each run, not by the simulation time.

Jones and Johnson emphasize the value of statistical design and analysis when the simulation time is long. Are the ideas only relevant when the simulator is slow? In discussing these methods with engineers, I am sometimes told that their simulator takes only seconds per run and that they can find solutions by simply making massive numbers of runs. If the only goal is to find 'the' optimum factor settings, I agree that these engineers can succeed quite well from exhaustive simulation. Often, though, other goals are also relevant, such as identifying the most important factors, characterizing *regions* of good response (and not just point optima), and finding robust operating regions (see Bates *et al.*[5], for a discussion). In all of those settings, good statistical design and analysis can play an important role even with fast run times. With many factors, as Jones and Johnson note, a sequential approach may be needed: first find the influential factors and then model their effects on the response. More work is needed on comparing design and analysis methods for the initial screening phase. Are LHC designs really more effective for screening than orthogonal array designs with a small number of levels for each factor?

The GASP approach has proven very effective in modeling data from computer experiments. In the example shown by Jones and Johnson, the GASP model with the LHC design clearly gives the best predictions. Other non-parametric regression methods are natural alternatives to GASP models. Neumann Ben-Ari and Steinberg[6] compared GASP models with projection pursuit regression[7] and MARS[8] on a suite of problems and found that GASP gave the best predictions.

As described by Jones and Johnson, the GASP approach models dependence of the response on the input factors entirely via the correlation function, as in Welch *et al.*[9]. The original formulation of this model by Sacks *et al.*[10] also included some regression terms (e.g. linear regression on each factor). Are the regression terms needed? Steinberg and Bursztyn[11] provided some useful insight. They showed that the dominant modes of variation captured by the correlation function in Section 2 of Jones and Johnson are in fact low-degree polynomial trends. Linear dependence, if present, can be represented very well by the correlation function, so no explicit linear regression terms are needed in the model. The same holds for quadratic trends or first-order interactions. Large values of θ_k in the correlation function will occur when there is strong linear dependence and need not correspond to non-smooth dependence, as stated by Jones and Johnson. Frequently, good surrogate models can be obtained with no regression functions. There is one important caveat. The correlation function will reflect smooth polynomial trends only within the range of the experimental input data; it will not extrapolate those trends outside the range of the data. If extrapolation is necessary, then inclusion of some additional ‘fixed effect’ regression functions is essential. An interesting compromise in this direction is the ‘blind kriging’ approach of Joseph, Hung and Sudjianto[12], which uses an initial GASP model to identify regression functions for inclusion in a final model.

In spite of the empirical successes of GASP, some words of caution are important. In particular, I want to stress that computer experiments, like all regression settings, can benefit greatly from statistical tools for probing high-dimensional data, both graphically and by modeling. Often transformations, either to the response or the predictors, can greatly improve predictions; don’t expect GASP to ‘automatically’ find them.

I will illustrate the above comments by reference to a recent collaboration to study migration of radionuclides into ground water from a nuclear waste repository. This was a ‘sensitivity analysis’ whose goal was to identify which site properties (lithology, soil distribution coefficients, etc.) were most influential in affecting the level of water contamination. The time frame was 10 000 years ahead; field data clearly cannot capture that scale and computer simulation is the only route for scientific study. Ranges for many of the factors in our study reflected wide scientific uncertainty as to their true values at the site. Further research to better determine the values of the factors was costly, so it was important to concentrate such effort only on the most influential factors. We used the RESRAD simulator developed at Argonne National Laboratories (http://web.ead.anl.gov/resrad/). Our experiment included 27 input factors and 900 observations from concatenating 3 LHC’s, each with 300 runs, an option built in to RESRAD. The results showed that there was *no migration at all* at 76% of the input settings. Migration at the remaining sites had a strongly skewed distribution. Efforts to directly predict the level of migration were heavily influenced by a small number of settings with very large responses. We decided, in consultation with the engineers on the team, that the analysis should focus on identifying the factors that were most highly predictive of having no migration at all, rather than focusing on the extent of migration; after all, eliminating migration is the desired response. Methods for classification thus replaced the GASP model as the main tools for analysis.

Computer simulation will continue to be a major component in many scientific and engineering investigations. GASP models, in conjunction with sound data analysis and exploration, can enhance our understanding of how the factors affect the response and can provide fast emulators. These methods will lead to useful solution of many problems. Those problems where the methods don’t work should be excellent sources for stimulating new ideas for design and analysis.

REFERENCES

1. Kenett RS, Steinberg DM. New frontiers in design of experiments. *Quality Progress* 2006; 61–65.
2. Thomke S. Enlightened experimentation: The new imperative for innovation. *Harvard Business Review* 2001; 65–75.

DOI: 10.1002/qre

3. Saltelli A, Tarantola S, Campolongo F, Ratto M. *Sensitivity Analysis in Practice*: *A Guide to Assessing Scientific Models*. Wiley: New York, 2004.
4. Cheng RCH, Kleijnen JPC. Improved design of queueing simulation experiments with highly heteroscedastic responses. *Operations Research* 1999; **47**:762–777.
5. Bates RA, Kenett RS, Steinberg DM, Wynn HP. Achieving robust design from computer simulations. *Quality Technology and Quantitative Management* 2006; **3**:161–177.
6. Neumann Ben-Ari E, Steinberg DM. An empirical comparison of Kriging with MARS and projection pursuit regression in modeling data from computer experiments. *Quality Engineering* 2007; **19**:327–338.
7. Friedman JH, Stuetzle W. Projection pursuit regression. *Journal of the American Statistical Association* 1981; **76**: 817–823.
8. Friedman JH. Multivariate adaptive regression splines. *Annals of Statistics* 1991; **19**:1–67.
9. Welch WJ, Buck RJ, Sacks J, Wynn HP, Mitchell TJ, Morris MD. Screening, predicting, and computer experiments. *Technometrics* 1992; **34**:15–25.
10. Sacks J, Welch WJ, Mitchell TJ, Wynn HP. Design and analysis of computer experiments. *Statistical Science* 1989; **4**:409–423.
11. Steinberg DM, Bursztyn D. Data analytic tools for understanding random field regression models. *Technometrics* 2004; **46**:411–420.
12. Joseph VR, Hung Y, Sudjianto A. Blind kriging: A new method for developing metamodels. *Journal of Mechanical Design* 2008; **130**:031102-1–031102-8.

DAVID M. STEINBERG
Department of Statistics and Operations Research
The Raymond and Beverly Sackler Faculty of Exact Sciences
Tel Aviv University
Tel Aviv 69978
Israel
E-mail: dms@post.tau.ac.il

DOI: 10.1002/qre

QUALITY AND RELIABILITY ENGINEERING INTERNATIONAL
Qual. Reliab. Engng. Int. 2009; **25**:545–547
Published online 18 June 2009 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/qre.1042

Discussion

Discussion (5): Jones–Johnson Paper

The paper by Jones and Johnson succeeds in its goal to provide an accessible introduction to the Gaussian Process model that has become widely used in the area of the design and analysis of computer experiments. Page limitations require that the paper be a broad-based overview. Here, I would like to make some additional comments, some of which arise from my own experience.

One area that the authors discuss is that of model calibration. Model calibration has developed in two slightly different directions. One is choosing parameters that are unique to the computer code (and have no counterpart in the physical process being studied) so that the computer code well approximates the physical data. This is what the authors discuss and it is sometimes referred to as tuning the computer code. An example of such a parameter might be the mesh size in a finite element model. The other direction is finding values of unknown physical parameters (for example, rate constants in chemical reactions) that appear in the code and whose values must be specified in the code. This is often done by finding the values that bring the code into agreement with physical data and assuming that these values are good estimates of these unknown physical constants. This is sometimes referred to as calibration. An important assumption is that the mathematical model on which the code is based is a very good description of the actual physical process, so that if the correct values of the physical constants are supplied to the code, the code will agree as closely as possible with the physical data. Unfortunately, in many applications I believe this assumption is not warranted because our mathematical models are at best only reasonable approximations. Furthermore, when there are many such unknown physical constants it may be that many combinations of values do an equally good job of making the code agree with physical measurements. The tuning problem is well-defined, but not the calibration problem. See Loeppky *et al.*[1] for more details.

I also point out that the references provided by the authors are generally Bayesian approaches to calibration. An earlier non-Bayesian approach can be found in the paper by Cox *et al.*[2].

The authors make the excellent point that there is a strong relationship between the design and the model used to fit the data. This is a topic that I have explored in some detail. When a Gaussian Process model is used, Marin[3] shows that, for purposes of prediction, there is little difference in the performance of various space-filling designs (maximin LHDs, uniform designs, good lattice point designs, scrambled nets, Niederreiter sequences, and Sobol sequences) when compared with a broad range of true response functions. However, space-filling designs generally outperform designs that are not space-filling, such as classical D-optimal designs for low-order polynomial models. This agrees with the finding of the authors in their example.

The authors mention design augmentation as a useful strategy. In my opinion, sequential experimentation (when possible) is superior to one-stage designs such as space-filling designs for fitting Gaussian Process models. The general approach is to develop a design criterion (sometimes called an improvement criterion) appropriate to the particular goal of the experiment and then use this criterion to choose designs points based on what one has learned from previous runs of the computer code. Some of the earliest papers employing sequential strategies are those for finding the optimum of the code: see Schonlau[4], Jones *et al.*[5], Williams *et al.*[6], and Huang *et al.*[7]. Jin *et al.*[8], Lehman[9], Fahrang-Mehr and Azarm[10], Marin[3], Ranjan *et al.*[11], Lam[12], Kumar[13], and Roy[14] investigate sequential strategies in a variety of contexts and, in general, these sequential approaches are superior to using a fixed one-stage design. This is not surprising because sequential approaches allow one to use what one has learned from previous runs about the functional form of the computer code output to decide where next to run the code.

Developing some intuition about the Gaussian Process model and the related predictor

$$\hat{y}(x) = \hat{\mu} + r'(x, \tilde{\theta}) R^{-1}(X, \tilde{\theta})(y - \hat{\mu} 1_n)$$

given in the paper can be difficult. In my experience, fitting these models (i.e. finding estimates of the model parameters and then computing the predictor given above) to a variety of test functions and comparing the results with those obtained using regression is a useful exercise and helps to develop such intuition. Section 4 of the paper is an example of such an exercise. Proc Mixed in SAS and JMP are two commercial software packages that allow one to fit Gaussian Stochastic process models. I have found R and Matlab useful for writing my own code to fit these models. Writing simple code and using it to fit test functions provides additional insight. Interesting test functions include a simple planar surface, polynomials, sinusoidal functions with high frequency, and surfaces that are flat around the boundaries but have a few central peaks. The latter surfaces do not look stationary and because the Gaussian Process model assumes stationarity, it is interesting to see how well one can fit such surfaces. One discovers that such test functions can often be fit quite well with sufficient observations. One also discovers that there are numerical issues that must be dealt with. For example, the likelihood can be quite flat and optimization algorithms may not converge. Also, the maximum likelihood estimates of the correlation parameters can produce a fitted model that is less than ideal. By trial and error one can often find estimates of the correlation parameters that provide better fit. Cross-validation is an alternative to maximum likelihood, as is some sort of penalized likelihood approach. See Li and Sudjianto[15]. Exploring the piston slap data in Li and Sudjianto[15] is a good exercise. Although not discussed in Li and Sudjianto[15], finding the correlation parameter estimates that minimize the cross-validation error produces results superior to those using penalized likelihood.

The correlation function

$$R_{ij}(X, \theta) = \exp(-\Sigma_k \theta_k (x_{ik} - x_{jk})^2)$$

given in the paper is sometimes referred to as the Gaussian correlation function. Some of the numerical problems arise from the exponential nature of the Gaussian correlation function. For large values of the correlation parameters (the Θ_k in the above equation), the correlation goes to 0 quite rapidly as points get farther and farther apart. For a design that has a few points clustered very closely together but the remainder more widely spread out, large values of the correlation parameters capture the local variation at the clustered points in the sense that if $(x_{ik} - x_{jk})^2$ is very small, $\exp(-\Theta_k (x_{ik} - x_{jk})^2)$ can take on a reasonable value (0.8 for example). But for points where $(x_{ik} - x_{jk})^2$ is not small, $\exp(-\Theta_k (x_{ik} - x_{jk})^2)$ will be nearly 0 and this can produce a fitted model that has a very 'spikey' shape away from the clustered points. One often encounters this problem when one begins to fit Gaussian Process models to test data in order to develop some intuition about the models. At the other extreme, when fitting the model to data lying on a plane, the correlation parameters are driven to 0 and this can lead to a correlation matrix that is near singular. One way to avoid some of these numerical problems is the following. Place bounds on the correlation parameters so that they cannot be too close to 0 and so that they are not allowed to be too large. Restrict the maximum likelihood search within these bounds. This can often produce a predictor that fits reasonably well. A Bayesian approach that uses these bounds in the priors for the correlation parameters may be effective. An alternative to the Gaussian correlation function is the cubic correlation function. See Santner *et al.*[16] for more on the cubic correlation. In my experience the cubic correlation function is better behaved (leads to fewer numerical problems) than the Gaussian correlation function. In addition, using the cubic correlation function produces a predictor that is a cubic spline. Practitioners may have better intuition about cubic splines than predictors based on the Gaussian correlation function.

An area requiring much more research is the use of Gaussian Process models when the number of explanatory variables is large. Numerical issues become a serious problem in this case as well as when there are a large number of observations. I suspect that in practice one is likely to encounter computer models having many explanatory variables. Factor screening can help if the computer model is relatively insensitive

DOI: 10.1002/qre

to a large proportion of the explanatory variables, because these can be fixed and only the few remaining explanatory variables are varied.

I again thank the authors for providing a readable introduction to Gaussian Process models in computer experiments. I hope this will stimulate readers to use and explore these models. This is a very interesting area with lots of open questions and a need for innovative methodology.

REFERENCES

1. Loeppky J, Bingham D, Sacks J, Welch W. Computer model calibration or tuning in practice. *Research Paper 2006-221*, Department of Statistics, University of British Columbia, Canada, 2006.
2. Cox DD, Park SJ, Singer CE. Statistical calibration of computer simulations. *Reliability Engineering and System Safety* 2001; **91**:1358–1363.
3. Marin O. Designing computer experiments to estimate integrated response functions. *PhD Thesis*, The Ohio State University, 2006.
4. Schonlau M. Computer experiments and global optimization. *PhD Thesis*, University of Waterloo, 1997.
5. Jones D, Schonlau M, Welch W. Efficient global optimization of expensive black-box functions. *Journal of Global Optimization* 1998; **13**:455–492.
6. Williams BJ, Santner TJ, Notz WI. Sequential design of computer experiments to minimize integrated response functions. *Statistica Sinica* 2000; **10**:1133–1152.
7. Huang D, Allen TT, Notz WI, Zeng N. Global optimization of stochastic black-box systems via sequential kriging meta-models. *Journal of Global Optimization* 2006; **34**:441–466.
8. Jin R, Chen W, Sudjianto A. On sequential sampling for global metamodeling in engineering design. *Proceedings of DETC 2002. ASME 2002 Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Montreal, Canada, September 2002.
9. Lehman JS. Sequential designs of computer experiments for robust parameter design. *PhD Thesis*, The Ohio State University, 2002.
10. Fahrang-Mehr A, Azarm S. Bayesian meta-modeling of engineering design simulations: A sequential approach with adaptation to irregularities in the response behavior. *International Journal for Numerical Methods in Engineering* 2005; **62**:2104–2126.
11. Ranjan P, Bingham D, Michailidis G. Sequential experiment design for contour estimation from complex computer codes. *Technometrics* 2008; **50**:527–541.
12. Lam CQ. Sequential adaptive designs in computer experiments for response surface model fit. *PhD Thesis*, The Ohio State University, 2008.
13. Kumar A. Sequential calibration of computer models. *PhD Thesis*, The Ohio State University, 2008.
14. Roy S. Sequential-adaptive design of computer experiments for the estimation of percentiles. *PhD Thesis*, The Ohio State University, 2008.
15. Li R, Sudjianto A. Analysis of computer experiments using penalized likelihood in Gaussian kriging models. *Technometrics* 2005; **47**:111–120.
16. Santner TJ, Williams BJ, Notz WI. *The Design and Analysis of Computer Experiments* (Springer Series in Statistics). Springer: New York, 2003.

WILLIAM NOTZ
The Ohio State University
Columbus, OH
U.S.A.
E-mail: win@stat.osu.edu

DOI: 10.1002/qre

QUALITY AND RELIABILITY ENGINEERING INTERNATIONAL
Qual. Reliab. Engng. Int. 2009; **25**:549–550
Published online 18 June 2009 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/qre.1043

Rejoinder

Design and Analysis for the Gaussian Process Model: a Rejoinder[‡]

We appreciate the thoughtful responses of all the discussants. They have extended our content in several useful ways.

Qian and Wu have provided a powerful methodology for extending the range of inquiry of computer experiments to include categorical factors. They show how to incorporate them in both design and analysis. We think this is impressive work and worthy of a standalone paper. We recommend that practitioners using this approach consider whether the change across the levels of a prospective categorical factor makes a fundamental change in the system being studied. If so, it might be preferable to run separate experiments. If, as in their example, the level changes do not require a major reprogramming of the simulator, then using their methods will result in substantial savings in computer time and the advantage of a global model instead of separate models for each combination of categorical factor levels.

The discussions by both Morris and Loeppky and Williams point out that the uncertainty intervals on unsampled points generated by using our 'plug-in' approach from maximum likelihood estimates may be too short. The full Bayes approach they advocate incorporates the uncertainty about the parameters in these intervals making them longer and more realistic. We look forward to implementation of the full Bayes approach in commercial software. In the meantime the GEM software is available through the web site referenced by Morris.

Most of the discussants point out that GASP models depend on stationarity. That is, the correlation between pairs of points the same distance apart in every coordinate does not depend on their location in space. If the response surface is dramatically different in one region then this assumption is violated and the resulting fitted GASP model may not perform well in that region. Notz, however, points out that given sufficient observations, the GASP model often performs adequately despite apparent nonstationarity. Nevertheless, the assumption of stationarity has implications in both design and analysis.

Suppose, the researcher suspects in advance that the response behaves in a dramatically different way in a region of the hypercube of interest. We think that the development of space filling designs for constrained factor spaces will be a fruitful area of future research. Can the Latin Hypercube design be modified to accommodate linear inequality constraints involving multiple factors? If, as Steinberg points out, the possible design points must lie on a coarse grid, then the ever popular Latin Hypercube design is not a viable alternative.

Often the nonstationarity of the response surface is discovered in the course of the analysis. Loeppky and Williams discovered some indication of nonstationarity in our example that models the F Quantile function. There is not yet any definitive methodology for dealing with this problem. A practical approach is to fit separate GASP models in the different regions. If there is not enough data in a region of interest, the augmentation approach of Loeppky and Williams may be useful. We also recommend their use of cross validation and model diagnostics to indicate possible problem areas in the design region.

We thought Morris's discussion of the relationship of maximin and maximum entropy designs was interesting and deserves further investigation.

[‡] This article is a U.S. Government work and is in the public domain in the U.S.A.

Published in 2009 by John Wiley & Sons, Ltd.

Steinberg points out that sometimes a computer simulation run is not expensive. In such cases, researchers sometimes generate thousands of runs in a brute force attempt to find an optimum. Fitting GASP models becomes painfully slow when the sample size is larger than a few hundred runs mentioned by Morris. The current estimation methods are limited by the need to invert nxn matrices. When $n > 10\,000$, this is infeasible. Should researchers be encouraged to generate less data that are more strategically placed? This begs the question of what to do if you are actually presented with such a data set. The development of an interpolating model that can handle large multidimensional data sets would be a huge contribution.

Large sample size is not the only problem in the analysis of GASP models. Notz also mentions the difficulty in fitting GASP models when there are many factors. Though the correlation function we introduced in the paper is probably the most popular, Notz advises that the cubic correlation function can often result in smoother fits and faster computation. This agrees with our own experience.

There are still differences of opinion about the value of including regression terms in the GASP model. Some authors prefer just fitting the mean and letting the correlation structure handle everything else. Others like to add linear or other fixed effect terms to the model. We would like to point out that the GASP model prediction goes to the mean when prediction is made far from the data in any direction. If there are strong reasons to prefer different asymptotic behavior, then this needs to be a part of the fixed effects model.

Steinberg points out that we should not forget everything we know about data analysis just because there is no stochastic component in deterministic computer models. Certainly, power transformations of the response may result in better fitting models leading to better intuition about how the process works. In situations involving functional responses, it may be necessary to do a substantial amount of preprocessing to avoid the tedium (not to mention dubious practice) of fitting a separate model to individual responses.

We thank the discussants for their many valuable insights. The various issues they raise emphasize that computer simulation experimentation is a wide open and vibrant research area. General solutions in the remaining problem areas are of vital importance because the use of computer simulation will continue to explode either with or without statistical input.

BRADLEY JONES
SAS Institute
Cary, NC 27513
U.S.A.

RACHEL T. JOHNSON
Naval Postgraduate School
Monterey, CA 93943
U.S.A.

Published in 2009 by John Wiley & Sons, Ltd.

DOI: 10.1002/qre