# Propagation of Computational Uncertainty Using the Modern Design of Experiments

**Richard DeLoach**
NASA Langley Research Center
4 Langley Boulevard
Hampton, Virginia 23681
United States of America

Richard.DeLoach-1@NASA.gov

## *ABSTRACT*

*This paper describes the use of formally designed experiments to aid in the error analysis of a computational experiment. A method is described by which the underlying code is approximated with relatively low-order polynomial graduating functions represented by truncated Taylor series approximations to the true underlying response function. A resource-minimal approach is outlined by which such graduating functions can be estimated from a minimum number of case runs of the underlying computational code. Certain practical considerations are discussed, including ways and means of coping with high-order response functions. The distributional properties of prediction residuals are presented and discussed. A practical method is presented for quantifying that component of the prediction uncertainty of a computational code that can be attributed to imperfect knowledge of independent variable levels. This method is illustrated with a recent assessment of uncertainty in computational estimates of Space Shuttle thermal and structural reentry loads attributable to ice and foam debris impact on ascent.*

## 1.0 INTRODUCTION

The quantification of uncertainty in a computational code is complicated by a number of factors that distinguish computational investigations broadly from the kinds of empirical studies that might be conducted in a laboratory or some other facility, such as a wind tunnel. The fact that replicated runs in a computational study display no variance is an especially relevant distinction that is often cited as an impediment to quantifying uncertainty in such studies.

This paper describes certain practical techniques for circumventing some of the features of computational investigations that complicate an assessment of uncertainty, including the absence of variance in replicated computations. These techniques are borrowed from a formal approach to experimentation that has been applied successfully in many non-aerospace industries throughout most of the 20$^{th}$ century, but which was introduced to elements of the experimental aeronautics community at Langley Research Center relatively recently, in 1997 [1]. Since then, researchers at Langley and elsewhere have applied formal experiment design techniques to an increasing array of aerospace applications. These techniques, taken as a whole, constitute a formal experimental research process known at Langley as the Modern Design of Experiments (MDOE). Three integrated elements of this process—design, execution, and analysis—comprise an extension of design of experiments methods commonly used in industrial engineering and elsewhere for product and process improvement. These extensions focus on special problems in aerospace research, including the prevalence of generally more complex response functions than are encountered in process and product improvement applications, extremely high precision requirements that are especially susceptibility to covariate effects, and the special importance of cycle-time reduction in aerospace research [2–4].

MDOE methods were introduced at Langley to respond to a perceived need to improve both quality and productivity in aerospace research generally, and in wind tunnel testing operations specifically. Productivity and quality enhancements that have been achieved by these methods have been documented for numerous wind tunnel tests [5–18]. MDOE methods have been particularly effective in applications with especially stringent quality requirements, such the calibration of ground test facilities and instrumentation, and in applications where significant resource constraints made high productivity testing a special priority [19–30].

The most recent focus of MDOE has been in computational experiments, where the size of case matrices that can be run for resource-intensive applications is often limited by cost constraints [31,32]. The complexity of such applications also makes it difficult to achieve unambiguous insights from a necessarily constrained volume of computed response estimates. The propagation of errors can be especially problematical given the practical constraints of a complex computational experiment. Ordinary Monte Carlo methods commonly used for this purpose can be quite resource intensive, and generally result in single-point solutions that cannot reveal broad patterns of error behaviour without a significant commitment of resources.

The next section of this paper presents a broad overview of the Modern Design of Experiments, followed by a section describing its application to the analysis of computational experiments that can be designed for uncertainty assessment as well as for achieving certain enhancements in productivity and cost control. Section 4.0 describes a variance partitioning method common to MDOE analyses that provides an objective metric of the degree to which a polynomial approximating function matches the underlying code for which it is intended to serve as a surrogate. Error propagation is also addressed in this section, in the context of applying the surrogate models developed from MDOE computational experiments for this purpose. Section 5.0 reviews a recent application of these methods to the assessment of uncertainty in computational codes designed to quantify thermal and structural reentry loads associated with ice and foam debris impacts to the Space Shuttle's Thermal Protection System on ascent. The paper concludes with a brief discussion of certain related uncertainty topics including limitations of the MDOE method for computational uncertainty assessment, and a summary of key points.

## 2.0 OVERVIEW OF MODERN DESIGN OF EXPERIMENTS

The Modern Design of Experiments begins with a premise that the only reason to conduct an experiment—physical or computational—is to learn something new about a system under study. This seemingly innocuous statement is the basis of most of what distinguishes MDOE from the dominant mode of experimental research in today's aerospace industry.

Aerospace experiments generally employ some variation of what is known in the literature of experiment design as One Factor At a Time (OFAT) testing. In an OFAT experiment, one independent variable is selected for investigation while all others are held at a fixed level. For example, in a wind tunnel test, the Mach number is typically held constant while the angle of attack (AoA) is varied systematically over some prescribed range. When the last AoA point has been acquired, the Mach number in incremented to a new level that is held constant while the same AoA levels are again set. This process continues until every combination of AoA and Mach number of interest has been examined. The same process can be easily extended, at least in theory, to as many independent variables as are of interest. In practice, for even a moderate number of independent variables, the data volume required for a complete OFAT experiment is generally too large to be accommodated by typical research budgets and/or time constraints. It is therefore common for OFAT researchers to leave a large number of independent variable combinations unexamined in deference to such constraints.

The OFAT method is fundamentally an exhaustive enumeration technique. Data are acquired at systematically examined combinations of independent variable levels, typically for as many such combinations of interest as the researcher can afford to examine. The principal product of such an OFAT investigation is perceived to be the data acquired in this way, so productivity and quality metrics are cast in data-centric terms. This perception lead quite naturally to a view common in the mid-1990s at Langley Research Center and elsewhere, that large-scale research facilities such as wind tunnels could be regarded essentially as *industrial factories* for the production of a particular product: *data*. Standard industrial process control techniques were invoked to assure a high-quality product by attempting to assign causes to sources of unexplained variance, with a view to eliminating them. Productivity was also cast in terms of data. The focus was on high data acquisition rates in order to maximize data volume. Thus, a high quality, high productivity experiment was considered to be one which produced the principal product, data, with high quality and in high volume.

MDOE challenges the fundamental notion of OFAT testing, which is that high-volume, high-quality data acquisition is the objective of experimental research. MDOE practitioners do not conduct experiments to "get data," but rather, as noted above, to learn something new about the system under investigation. This change in viewpoint has profound implications for both productivity and quality. If data is no longer perceived as the principal product of experimental research, why maximize data volume and data quality? Why focus on the quality and volume of what is *not* the product?

From the MDOE perspective, it is *knowledge* that we seek from an experiment, not *data*. We wish to "know" the system under study, which for practical purposes means that we wish to be able to adequately predict its future behavior for any combination of independent variable levels of interest. In order to achieve this general objective, it is necessary to achieve a number of specific objectives, each of which involves making some inference about the system in order to answer a specific question about it. We may wish to know if the new wing provides more lift than the old wing at cruise, for example, or if the elevator authority is influenced by aileron deflections. We often wish to develop mathematical models capable of predicting various system responses as a function of independent variable levels. In those cases, we may wish to know whether each term in a candidate regression model is statistically significant or not. Each of these examples entails making one inference or another; and it is these inferences, rather than the raw data points upon which they are based, that are regarded as the principal product of an MDOE experiment.

Because inferences are regarded as the product of an MDOE experiment, MDOE concepts of productivity and quality are cast in terms of these inferences. From the MDOE perspective, a productive experiment is not necessarily one that generates large volumes of data (which simply increases costs), but rather is one that allows us to make the inferences necessary to adequately answer whatever questions motivated the experiment. (This implies that the researcher *knows* what questions motivated the experiment, which is a whole other discussion! The fact that MDOE methods motivate such awareness among test personnel is regarded by its practitioners as one of its principal virtues.)

The MDOE view of productivity differs diametrically from the conventional OFAT view, in that MDOE practitioners regard extremes in data volume are a sign of inefficiency rather than a metric of productivity. One reason is that the value added by each successive data point is a monotonically decreasing function of the volume of data already in hand. (One more data point is likely to add significant value when all you have so far is a small number of points. On the other hand, while it is always better to have a million and one points than a million points, it is not much better.) Since the value added by each new point is less than the value added by the previous point, there is ultimately a volume of data beyond which the value added by the next data point must fall below the cost of acquiring it. Such a point of diminishing returns argues against the OFAT notion that "more is always better" when it comes to data volume.

The fact that the value added by each new data point is relatively great when little data are yet in hand, and relatively little after a critical mass of data has been acquired, is illustrated in Figure 1. This figure describes a landing dynamics experiment recently designed at Langley Research Center using MDOE. Each run consists of a unique combination of aircraft landing speed, runway material, and tire load (proportional to aircraft weight), for which stopping distances were measured. "Value" is cast in this figure in MDOE terms by representing it as the probability of a correct inference. Note that the early data points are responsible for most of the increase in value, with later points adding relatively little at the margin (the curve approached 100% asymptotically but never achieves it). In this case, one could state conclusions with 99 % confidence after 11 runs. If this is consistent with the researcher's inference error risk tolerance, there is little need to execute more runs than this.

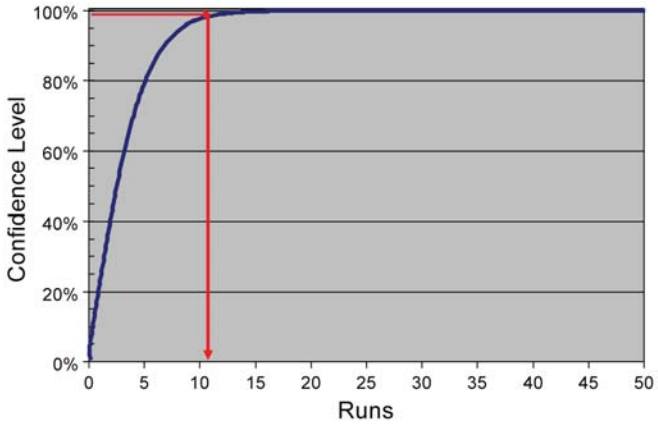


**Figure 1: Value added by each new run in a landing dynamics experiment. 99% confidence could be achieved in 11 runs.**

The volume of data at which the knee in Figure 1 occurs will depend on a number of factors related to precision requirements and inference error risk tolerance; but the same general behavior is common to all experiments, including both computational and physical experiments. That is, there is a knee in this curve that is generally encountered after fewer runs than one might anticipate, and generally after many fewer runs than are typically specified in an OFAT test matrix, for which high data volume is equated with productivity. This behavior is routinely exploited in both physical and computational MDOE experiment designs to achieve test objectives with compact run schedules that minimize direct operating costs as well as cycle time. That is, the key to MDOE productivity is to know how many runs to execute, and to stop expending the resources required for further data acquisition when this volume of data is in hand. (Numerous tests are available to determine if initial data volume estimates were adequate.) This enables us to design very compact experiments to support the computational uncertainty assessment methods described in subsequent sections of this paper.

The MDOE philosophy of testing differs from conventional OFAT testing with respect to quality as well as productivity. A quality MDOE experiment does not require low levels of unexplained variance in the raw data, but rather a low probably of error in each of the inferences that are drawn from the data. From the MDOE perspective, "good data" (data with little unexplained variance and convenient distributional characteristics), while always desirable, is a cost issue rather than a quality issue. Inference error probabilities can be made to approach zero arbitrarily closely by acquiring a sufficient volume of data, but more data (and hence greater costs and extended cycle times) are required for lower-quality data than higher-quality data. In

the context of Figure 1, lower quality data simply shifts the knee in the curve to the right. But regardless of the quality of the data, if a sufficient volume is acquired (that is, if one goes far enough to the right in Figure 1), it will be possible to make inferences with a level of confidence that satisfies any specified inference error risk criterion that can be achieved with a finite volume of data. It is not possible with a finite volume of data to ever achieve 100% confidence in any inference (inference error probability of zero), but this ideal can be approached asymptotically, as Figure 1 illustrates.

To summarize the difference between OFAT and MDOE quality and productivity concepts, the OFAT practitioner seeks quality data in high volume, while the MDOE practitioner seeks answers to specific questions with a high probability of being right. The OFAT approach tends to defer the hard work of analysis, focusing instead initially on acquiring enough data to answer whatever questions might be formulated in the future. Unfortunately, in practice it is very difficult to anticipate all future questions that may be of interest. The MDOE practitioner develops compact experiments designed to answer specific questions, and relies upon an objective test exit strategy based on stopping when enough data are in hand to assure that specified inference error risk tolerances are satisfied. (Note that the MDOE focus on "answers to specific questions" is not incompatible with a desire to achieve a broad predictive capability for a given system under study. Typical MDOE inferences focus on whether specific terms in a general response prediction model are statistically significant or not, for example. While such inferences relate to specific model coefficients, the result is a model that provides a general predictive capability.)

MDOE experiments feature an exit strategy that places a strong emphasis on uncertainty assessment, which is the central element of MDOE's inference error risk management approach to experimentation. This integration of uncertainty assessment into the stopping criteria for an experiment has the practical effect of significantly reducing costs and cycle time—often by a factor of two or more. In brief, the MDOE practitioner knows when the answers are reliably in hand and is then able to stop expending resources, while the OFAT practitioner tends to stop only after all available resources are exhausted. This difference can provide the MDOE practitioner with a significant competitive advantage that is cumulative over time.

## 3.0 DESIGN OF MDOE COMPUTATIONAL EXPERIMENTS FOR UNCERAINTY ASSESSMENT

In this section, we describe how MDOE methods are applied to design a computational experiment that can lead to the development of a surrogate model for some complex underlying computational code. This surrogate can then be applied to propagate uncertainty in the input parameters of the underlying code. An analysis of the unexplained variance in the computational experiment may also provide insights into the intrinsic uncertainty associated with the underlying code.

### 3.1 Graduating Functions

We seek to describe selected system responses in terms of the levels of various independent variables or "factors," to use common terminology from the experiment design literature. Our objective is always to be able to predict system behavior in terms of these factor settings. So, for example, for a specified set of flight state variable levels (Mach number, Reynolds number, etc.) and vehicle attitude and configuration variable levels (angles of attack and sideslip, and control surface deflections, etc.), we wish to be able to adequately predict such system responses as forces, moments, pressures, temperatures, and so forth. We define a response prediction as "adequate" if there is an acceptable probability that such a prediction is within specified tolerances for any arbitrary combination of factor levels that are all within some range of interest

that was considered in the study of the system. It is the researcher's responsibility to demonstrate this objectively, in which case he is entitled to say that he "knows," or "has knowledge of," the system.

Except in special cases in which the underlying physics is especially well understood, a closed-form relationship between system response variables (say, the forces and moments of interest in an experimental aeronautics study) and factors that influence system response (angle of attack, Mach number, and various control surface deflections, for example) is unavailable. We can write down a general representation of this relationship, however, using a Taylor series.

The familiar Taylor series represents a function in the neighborhood of some reference point as an infinite series, each term of which depends upon two quantities: a derivative of the function evaluated at the reference point (call that point $\boldsymbol{a}$), and the displacement from that reference point. Let the magnitude of a vector, $\boldsymbol{r}$, define the radius of a neighborhood within which the function is expanded with a Taylor series.

Some functions cannot be written as a Taylor series because they have some singularity, so that not all of their derivatives exist. In other circumstances, the response of a system may be so complex, even over highly constrained factor ranges, that an impractical number of terms must be retained in a Taylor series to adequately represent it. However, it is fortuitous for our purposes that many real-world physical response functions can be represented as a Taylor series over practical ranges of the independent variables.

It is particularly convenient to employ the Taylor series to represent *analytic* functions, defined as those for which this series not only exists but converges for every $\boldsymbol{x}$ within a hypersphere centered at $\boldsymbol{a}$ with a radius of $|\boldsymbol{r}|$, and for which the sum of all the terms in the series equals $f(\boldsymbol{x})$, the true value of the function at any $\boldsymbol{x}$ within this interval. A function is analytic if and only if it can be represented as a power series, in which case the coefficients are necessarily those of a Taylor series. Real-world response functions are commonly analytic, and can therefore be represented by such a power series.

For illustration, let $y$ be an unknown mathematical function of two variables $\xi_1$, $\xi_2$, which we would like to represent as a Taylor series in the neighborhood of $\xi_1 = a_1$ and $\xi_2 = a_2$. Perhaps $y$ is pitching moment, which depends on angle of attack and Mach number in ways we understand generally, but are not yet able to predict with adequate precision, and $a_1$ and $a_2$ are cruise AoA and cruise Mach. In that case, for Mach numbers and angles of attack in some neighborhood around cruise, we can represent pitching moment to second order with the following Taylor series, in which we have dropped terms of third order and higher:

$$\begin{aligned} y(\xi_1,\xi_2) = y(a_1,a_2) &+ \left.\frac{\partial y}{\partial \xi_1}\right|_{\xi_1=a_1}(\xi_1-a_1) + \left.\frac{\partial y}{\partial \xi_2}\right|_{\xi_2=a_2}(\xi_2-a_2) \\ &+ \frac{1}{2!}\left[\left.\frac{\partial^2 y}{\partial \xi_1^2}\right|_{\xi_1=a_1}(\xi_1-a_1)^2 + \left.\frac{\partial^2 y}{\partial \xi_1 \partial \xi_2}\right|_{\xi_1=a_1,\xi_2=a_2}(\xi_1-a_1)(\xi_2-a_2) + \left.\frac{\partial^2 y}{\partial \xi_2}\right|_{\xi_2=a_2}(\xi_2-a_2)^2\right] \end{aligned} \tag{1}$$

It is not possible to evaluate the derivative analytically, nor can we assess the first term in the series without knowing the function, $y$. However, let us introduce a convenient change of variables:

$$x_i = \frac{\xi_i - \tfrac{1}{2}(H_i + L_i)}{\tfrac{1}{2}(H_i - L_i)} \tag{2}$$

or

$$\xi_i = \tfrac{1}{2}\left[ H_i\left(x_i + 1\right) - L_i\left(x_i - 1\right)\right] \tag{3}$$

where $L_i$ and $H_i$ are the lower and upper limits on intervals centered on the $a_i$, throughout which the Taylor series is to be evaluated. Dropping terms of order three and higher results in errors in this series approximation that can be driven arbitrarily close to zero by selecting $L_i$ and $H_i$ sufficiently close to $a_i$, the center of the interval for variable $\xi_i$. The variables $x_i$, commonly called coded variables, have certain practical properties. They range between ±1 regardless of the units of their corresponding physical variables, $\xi_i$, and so present less potential for round-off errors during floating-point regression analyses. This consistency in range also helps in interpreting the results of regression and ANOVA computations. The coded variables are also centered, which decouples slope and intercept terms in a regression analysis.

After introducing this change of variables and certain notational changes, the Taylor series reduces to a simple polynomial function as follows:

$$y\left(x_1, x_2\right) = b_0 + b_1 x_1 + b_2 x_2 + b_{12} x_1 x_2 + b_{11} x_1^2 + b_{22} x_2^2 \tag{4}$$

where the $b_i$ are constants related to the derivatives evaluated at $\xi_i = a_i$. These constants are typically evaluated by regression, with a vector of measured responses fitted to an ensemble of independent variables. The resulting low-order polynomial is commonly called a *graduating function*, which Box and Draper [33] have evocatively described as a "mathematical French curve" because of its ability to adequately fit some limited range of a complicated response graph. For a graduating function in $k$ independent variables, we use the term "*inference space*" to denote a $k$-dimensional space in which each axis corresponds to one of the variables. Each unique location or *site* in this inference space represents a unique combination of independent variable levels. In principal and in practice, even the most complex response functions can generally be represented adequately by a piecewise continuous function consisting of contiguous low-order graduating functions that span the inference space of interest in a patchwork quilt of subspaces.

This polynomial series representation is important because it means that a finite number of terms representing a (sometimes severely) truncated series can be used to approximate response function values quite closely over sufficiently small intervals. In experimental aeronautics, these functions seldom need to be of order higher than about four to fit practical ranges of typical independent variables. Often third-order models are adequate, and occasionally a second-order model is sufficient. The details depend, of course, on the specific response variable and the range of independent variables. As is generally the case, experience is the most reliable guide in anticipating the order of the graduating function for each subspace.

## 3.2 Estimating the Response Function

We start with a general Taylor series representation of an unknown multivariate function. We can generally describe quite complex behavior with a finite number of low-order polynomials if each is fitted over a suitably constrained range of independent variables. It is not necessary to understand the underlying processes that produce the response of the system in order to describe it in this way.

The numerical values of the coefficients of such mathematical models can be estimated by ordinary regression methods from a sample of data consisting of system responses corresponding to specified factor levels. In both physical and computational research, direct operating costs and cycle time are dependent on the sample size. We therefore seek the smallest such sample adequate to estimate the coefficients.

A full $d^{th}$-order polynomial in $k$ factors features $p$ coefficients including the intercept term, where

$$p = \frac{(d+k)!}{d!k!} \tag{5}$$

Since one degree of freedom is associated with every coefficient in the model, this represents the fewest number of points that must be included in a sample to be used for this purpose.

It is generally prudent to specify additional degrees of freedom to allow for the fitting of higher-order models should the anticipated order be inadequate. These so-called lack-of-fit degrees of freedom (LOF df) consist of unique combinations of factor levels. An example will clarify how to objectively quantify in advance the smallest volume of data necessary to represent a response via piecewise continuous polynomials.

Assume as an illustration that some complex CFD code exists for an experimental aircraft and that it is to be used to generate a database of forces and moments over a range of -10° to +25° in angle of attack and -10° to +10° in angle of sideslip, for a Mach number range of 0.74 to 0.96. While the details will vary from test to test, it would not be atypical to specify an AoA step size of 1°, a sideslip step size of 2°, and a Mach number step size of 0.02. This would require a total of 16 × 11 × 13 = 2,288 CFD computations.

Compare this with the number of computations required with polynomial approximations to the underlying code. Assume that for either positive or negative angle of sideslip we anticipate that a $4^{th}$-order polynomial will be adequate to approximate each force and moment over the full subsonic Mach range of interest in each of three AoA ranges—pre-stall, stall, and post-stall. In such a case, a total of six $4^{th}$-order polynomials would be adequate to represent all forces and moments over the entire inference space of interest. Since these polynomials would each feature three independent variables, a minimum of 7!/(4!3!) = 35 points would be required for each of them by Equation (5), or a total of 6 × 35 = 210 computations. (The reader is entitled to ask how it is possible to know in this case that a $4^{th}$-order polynomial is adequate to account for all the complexities in unknown response functions of interest. We will have more to say about this in the next section; but suffice it to say for the present that while we rely on experience and subject-matter expertise to make initial estimates, we also rely upon numerous tests that exist to assess the adequacy of those initial estimates.)

Even allowing for additional LOF df or perhaps an initial higher-order representation in the post-stall region, the data volume requirement in this example compares quite favorably with the exhaustive enumeration approach to generating a database, requiring an order of magnitude fewer computational runs in this example. As an added bonus, the polynomial approximation technique allows responses to be estimated for any arbitrary combination of independent variable levels, not just the discrete combinations specified in the computational test matrix.

Results will vary depending on the complexity of the experiment, but in general the greater the number of independent variables, the greater the productivity advantage of the polynomial approximation method. A relatively small number of case runs can usually generate an adequate representation of a complex underlying code.

This polynomial approximation technique has implications for quality assessment in a computational experiment, in addition to the productivity enhancement and cost reduction advantages that have been illustrated in this example. We will describe these quality assessment implications in Section 4.0.

### 3.3 Inference Space Site Location for Quality Enhancement

We have seen that a low-order polynomial can be used as a surrogate model for a complicated underlying computational code when that underlying code is relatively expensive and time-consuming to execute. We have also seen how to perform an initial scaling of an experiment that can be conducted in order to generate such a surrogate model. Scaling is the process by which we estimate the smallest volume of data required to fit a polynomial that we initially believe (based on experience and subject matter expertise) will be adequate to approximate some underlying process (in the case of a physical experiment) or some underlying code (in the case of a computational experiment). The adequacy of such polynomial approximations is constantly tested throughout the experiment, and the original test matrix is augmented (additional data points added) as necessary to fit an alternative model.

The full scaling and augmentation processes is beyond the scope of this paper (consult reference [3] for more detail), but the volume of data predicted by Equation (5) generally represents a minimum, which will have to be modified to account for precision requirements and specified inference error risk tolerances, as well as an estimate of the intrinsic variance of the data [2]. Nonetheless, the general trend remains the same: relatively few runs with the underlying code will be sufficient to develop an adequate polynomial approximation to that code, which can then be used to inexpensively and rapidly estimate responses for a large combination of factor levels.

While the *average* prediction variance for a polynomial graduating function depends only on the number of parameters in the model, the intrinsic uncertainty of the environment, and the volume of data acquired [33], the distribution of that variance is generally not uniform throughout the inference space, and can be influenced—optimized—by the process of *site selection*. A site in an inference space simply represents a unique combination of factor levels, as noted above. Every data point in a test matrix corresponds to a particular site in the inference space. Once the experiment has been scaled; that is, once the total volume of data has been defined, we turn our attention to site selection—specifying which particular combination of factor levels to set.

The role that the data *volume* plays in the reduction of experimental uncertainty in physical experiments is broadly understood, at least qualitatively, but the roll that data *selection* plays in assuring research quality is not nearly so well understood by many aerospace researchers. Indeed, the fact that data selection is even a factor in influencing the quality of regression models is not universally recognized. Nonetheless, for a given volume of data, the specific combination of independent variable levels chosen in a test matrix does have an effect—often a profound effect—on the uncertainty in predictions that can be made with models developed from the data.

We use site selection as a means of instilling certain desirable properties in the distribution of prediction variances throughout the inference space. For example, we generally prefer this distribution to be broad, low, and uniform throughout a large segment of the inference space around its center, which is the point of special interest about which the Taylor series approximation to the underlying (unknown) response function was originally expanded. We can achieve this by making site selections that drive relatively larger prediction variances into the less-interesting corners of the inference space so that they are smaller near the center. It is also generally desirable for the distribution of prediction variances to be symmetrical. If the prediction variance depends only on the distance from the center of the inference space and not the direction, we say the variance distribution is *rotatable*. We can also make site selections that minimize the uncertainty in estimates of the individual regression coefficients.

Figure 2 illustrates how point selection influences quality for even the trivial case of a first-order function of one variable, which only requires two points to estimate the response function. Uncertainty in the response estimates translates into uncertainty in the response model, with the amount of uncertainty depending only on the separation of the data points in this simple case. While this figure applies to a very simple special case, the concept it illustrates is quite general. No matter how high the order of the model or how many independent variables, the distribution of points used to fit the response model will have an effect on the quality of the model that is produced. This is because system response estimates, whether measured in a physical experiment or calculated in a computational experiment, will feature some uncertainty, and the leverage that the uncertainty in each point has in influencing errors in the prediction model will depend on where it is located in the inference space.

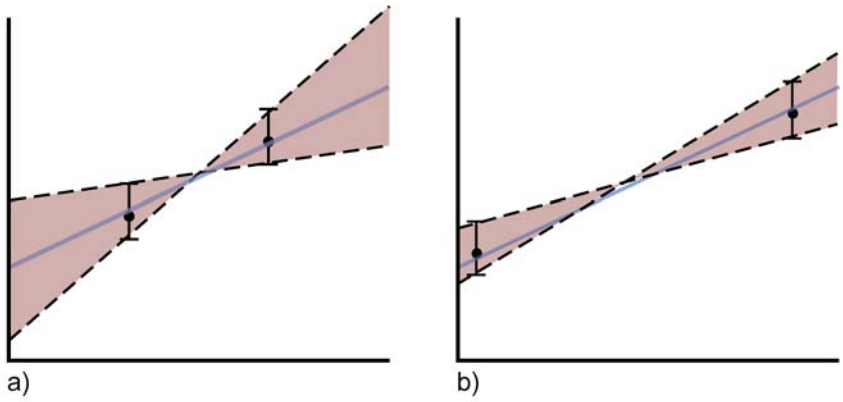


**Figure 2: Uncertainty in slope and intercept due to experimental error.**
**a) Smaller separation between points, b) larger separation between points.**

For complex reasons rooted in Group Theory, the run matrices that generate response surface models with the smallest prediction uncertainty in physical experiments tend to display a certain intrinsic symmetry. This symmetry is evident when the run matrix is represented graphically in an inference space using normalized or coded variables that range within ±1 over the range of corresponding physical variables. See Equation (2). Figure 3 illustrates a simple but common two-variable experiment design in variables coded by Equation (2), which is optimized to fit $2^{nd}$-order response functions. This design, known as the orthogonal Box-Wilson or Central Composite Design, features most of its points arranged in a circle, with each such point the same distance from the center of the design. More complex designs may feature a different distribution of independent variable levels throughout the inference space, but will generally still display some degree of symmetry.

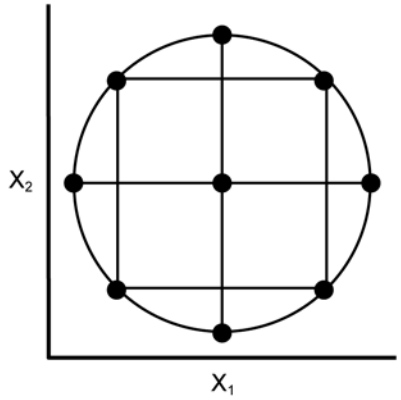


**Figure 3: Box-Wilson or Central Composite Design.**

It is not uncommon for aerospace applications to feature complex response functions of many variables. In such circumstances, a so-called D-optimal design strategy is often implemented in an MDOE experiment to make site selection decisions. A D-optimal design is one in which sites are selected to minimize the variance associated with the estimates of model coefficients. The process can be described in terms of the *design matrix*, **X**, which is an extension of the familiar test matrix. It has rows for each point in the design just as a conventional test matrix does; but it also has columns for every term in the response model, including the intercept term. The columns corresponding to first-order terms in the model contain the usual test matrix entries—the values of the corresponding independent variables that will be set in the experiment. Columns for higher-order terms in the model are constructed from the first-order columns by multiplication. Elements of the $x_1^2$ column are generated by simply squaring the corresponding elements in the $x_1$ column, for example. Elements in the column for the $x_1x_2$ interaction term are created by multiplying the corresponding elements in the $x_1$ and $x_2$ columns, and so on. The column corresponding to the intercept term $x^0$ is filled with 1s.

The covariance matrix, **C**, is a ($p \times p$) square matrix, where as before, $p$ (Equation (5)) is the number of terms in the polynomial including the intercept term. The covariance matrix is computed by pre-multiplying the design matrix by its transpose, inverting the product, and multiplying each element of the resulting matrix by the unexplained variance of the residuals, $\sigma^2$: $\mathbf{C} = (\mathbf{X'X})^{-1}\,\sigma^2$. It can be shown that the diagonal elements of the covariance matrix represent the variance in estimates of the regression coefficients. That is, the variance in the $i^{\text{th}}$ regression coefficient is simply $\mathbf{C}_{ii}$.

To minimize the uncertainty in the regression coefficients, we must minimize the determinant of the covariance matrix. This minimizes the confidence *ellipsoids* for each regression coefficient, the multidimensional equivalent of minimizing confidence *intervals* about a measured data point. The D-optimal design algorithm selects the subset of points to do this from a list of candidate points. Standard references describe the D-optimal algorithm in more detail [33,34].

It is commonly believed that site selection considerations are less important in computational experiments than in physical experiments, because there is no variance in a sample of replicates from a computational experiment. The fact that computational replicates feature no variance sometimes leads to a somewhat imprecise and misleading assertion that “there is no experimental error” in a computational experiment. A more careful framing of the statement would simply note that experimental error cannot be *estimated* by replicating points in a computational experiment. (This is analogous to a physical experiment featuring

single-point samples. Absent replicates in such an experiment, there is no apparent unexplained variance in the individual data points, but this does not imply that the measurements are error-free.)

It has been argued (e.g., by Giunta et al. [35]) that space-filling designs that invoke quasi-Monte Carlo sampling, orthogonal polynomial sampling, or Latin hypercube sampling are more appropriate in computational experiments than designs employed in physical experimentation to ameliorate the effects of random error by pushing test points toward the boundary of the design space, as in Figure 3. Figure 4 illustrates this graphically. The left of this figure illustrates a site selection strategy based on Figure 2, in which points are chosen to be near the boundaries of the design space in order to minimize the effects of random experimental error. The figure on the right illustrates a site selection strategy that might be adapted if all of the error in each data point were systematic. In this case, one would prefer to more evenly distribute sites within the design space, rather than concentrating them near the edges.

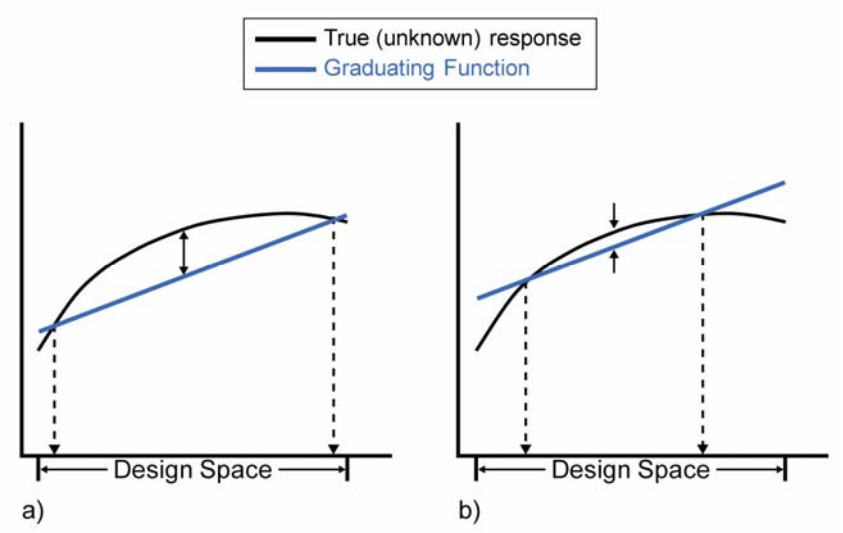


**Figure 4: Site selection. a) In designs that minimize pure error effects, the distance between points is relatively great. b) Designs that minimize lack-of-fit error fill the design space more uniformly.**

A generally uniform, space-filling design strategy may be appropriate if the system under study is to be described entirely in terms of discrete factor combinations that are intended to directly generate tabulated data for an aircraft database, for example. If, on the other hand, the purpose of the computational response estimates is to provide data that will be fitted to a low-order polynomial regression model, then a site selection strategy that minimizes the effect of errors at each site may be more appropriate.

There are truncation errors associated with representing an infinite Taylor series with a low-order polynomial approximation that neglects all terms of a certain order and higher. These errors are distributed among the coefficients of the retained lower-order terms. Regardless of the distribution of errors in those individual coefficients, the errors in any ensemble of polynomial response estimates will be normally distributed because those estimates are comprised of the sum of contributions from each term in the polynomial, a condition which the Central Limit Theorem guarantees will result in a normal distribution of errors.

We distinguish between the analyzed results of an MDOE computational experiment and the ensemble of response computations that represents the raw data in such an experiment. Any errors in the individual response estimates generated by the underlying code will in fact be systematic and infinitely repeatable, so that no sample of replicates will display any variance, no matter the sample size. While these features of computational response estimates complicate the task of characterizing the error distribution for the data, they neither imply that there is no error, nor that that the error does not follow some distribution. In fact, insofar as the error associated with any one computational response estimate can be assumed to consist of the algebraic sum of multiple errors associated with the development of the code, it is reasonable to assume by the Central Limit Theorem that the errors in an ensemble of computational data are in fact normally distributed.

In an MDOE computational experiment, we fit the response estimates made at strategically selected sites in the inference space to a candidate response model. The uncertainty that exists in each of the response estimates used to construct the model, whether easily revealed or not, has some impact on the quality of response predictions that can be made with the fitted model. The leverage that the error at each regression point has in generating uncertainty in the response predictions will depend on its location in the inference space. It is for this reason that inference-space site selection is an important quality assurance tactic in computational experiments, and why site-selection strategies which emphasize data acquisition near the edges of the inference space may in fact be more effective experiment designs for generating high-precision polynomial approximations than the space-filling strategies that may be more appropriate when the underlying code is used directly to populate response databases.

## 4.0 ERROR AND ERROR PROPAGATION IN COMPUTATIONAL EXPERIMENTS

Figure 5 illustrates a case in which the fit of an initial lower-order response model was found to be inadequate so that a higher-order model was evaluated. The lack-of-fit component of the initial fitting error was large compared to the pure error, indicating the need to fit a higher-order model. In this section, we will describe how to estimate pure error in a computational experiment, notwithstanding the fact that replicates display no variance in such an experiment. These techniques will be illustrated in Section 5.0 with a specific application.

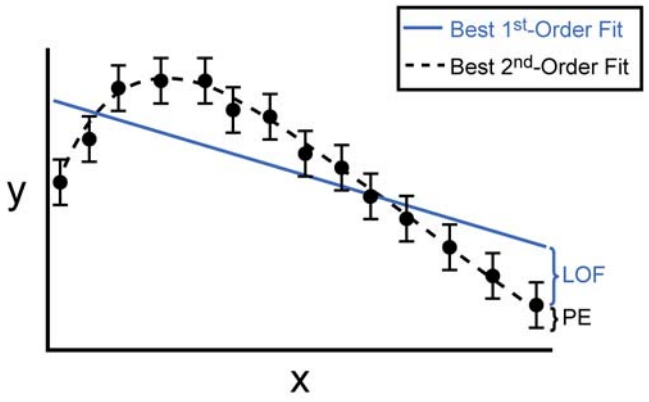


**Figure 5: Lack-of-Fit error in best 1st-order exceeds Pure Error, indicating a need for a higher-order fit.**

We have described in the previous section how a relatively small sample of computational runs can result in an adequate representation of responses of interest over some prescribed range of independent variables. We begin with the smallest number of points required to fit a polynomial of specified order in a given number of independent variables. We may also specify additional LOF df to accommodate more complex fits if this proves necessary.

In a physical experiment, such LOF df are commonly augmented by pure error degrees of freedom (PE df) consisting of genuine replicates of selected points in the test matrix. The PE df allow for a model-independent estimate of the intrinsic variance of the measurement environment, which is useful in assessing experimental uncertainty in physical experiments. Of course, in a computational experiment, PE df are ineffective for this purpose because any sample of replicates is variance free. (There is an exception in those cases for which experimental error is explicitly simulated, for example by Monte Carlo methods.) However, insofar as each response estimate in a computational experiment is the product of codes created by the fallible hand of man, we are on safe ground in assuming that it features some degree of error, however complicated it may be to estimate that error. That is, even in a computational experiment there is unexplained variance.

## 4.1 Unexplained Variance in Computational Experiments

We now introduce a strategy for assessing error in a computational experiment, notwithstanding the fact that replicated response estimates have no variance. We begin with an analogous situation from physical experimentation, in which we assess the error in data samples featuring no replicates.

Single-point samples (i.e., data points that are not replicated) are common in physical aerospace experiments such as wind tunnel tests that are conducted using OFAT methods, because time constraints result in a higher priority for new set-point combinations than for replicates of existing data. It has been noted above that while there can be no direct estimate of variance in physical experiments when there are no replicates, this does not mean that there is no experimental error. It simply means that the experimental error must be estimated by some other means than replication. Similarly, just because it is not possible to rely upon replication to produce a direct measurement of experimental error in a computational experiment, this does not mean that there is no error present or that it cannot be estimated in some other way besides replication.

In physical experiments for which there has been no replication, it is still possible to assess the unexplained variance in an ensemble of data by partitioning the total variance into explained and unexplained components. A regression model is used to explain as much of the total variance as possible, and any residual variance is defined as unexplained. The unexplained variance contributes to the uncertainty in response predictions that are made by the regression model.

If there is no significant systematic component in the residual variance and the errors in each point are independent, then the residual variance from the curve fit is expected to be an unbiased estimator of the pure error population variance. It is therefore possible to estimate the variance without ever replicating a single point. The only requirement is that there be more data points acquired than the number of regression coefficients in the model that will be fit to the data, to ensure that there are residual degrees of freedom available to assess the unexplained variance.

The mechanics of this technique can be applied just as easily to a computational experiment as to a physical experiment. Assume that we fit *n* data points to a model of the following form:

$$\hat{y}_i = \mathbf{x_i b} \tag{6}$$

which is just a generalization of Equation (4) expressed in a more compact vector-matrix notation. Assume that there are *m* unique points and that there are therefore *n-m* replicates. (We describe here the general case, which can be applied to physical experiments with genuine replicates or to computational experiments with no replicates, in which case *m=n*). Let $r_i$ be the number of replicates of the $i^{th}$ unique point, with $r_i=1$ for non-replicated points as in a computational experiment. If $y_{iu}$ is the $u^{th}$ value of the dependent variable measured at the $i^{th}$ unique site in the inference space, then the average of all the data is:

$$\overline{y} = \frac{\sum_{i=1}^{m}\sum_{u=1}^{r_i} y_{iu}}{n} \tag{7}$$

and the mean response at a given site is

$$\overline{y}_i = \frac{\sum_{u=1}^{r_i} y_u}{r_i} \tag{8}$$

We begin by computing the total variance of the data sample, which is defined as the total sum of squares divided by the total degrees of freedom, given the mean. The total sum of squares is computed as follows:

$$SS_{Total} = \sum_{i=1}^{m}\sum_{u=1}^{r_i}\left(y_{iu} - \overline{y}\right)^2 \tag{9}$$

There are *n-1* degrees of freedom associated with the total sum of squares, so the total variance is simply this:

$$\sigma^2_{Total} = \frac{\sum_{i=1}^{m}\sum_{u=1}^{r_i}\left(y_{iu} - \overline{y}\right)^2}{n-1} \tag{10}$$

We can partition the total sum of squares into a regression sum of squares and a residual sum of squares, as follows, where $\hat{y}_i$ is the response predicted by a graduating function that approximates the response (measured in a physical experiment or calculated in a computational experiment) for the $i^{th}$ data point:

$$SS_{Regression} = \sum_{i=1}^{m} r_i\left(\hat{y}_i - \overline{y}\right)^2 \tag{11}$$

and

$$SS_{Residual} = \sum_{i=1}^{m} \sum_{u=1}^{r_i} \left( y_{iu} - \hat{y}_i \right)^2 \tag{12}$$

The sums of squares are additive; that is, $SS_{Total} = SS_{Regression} + SS_{Residual}$, so the residual sum of squares is often computed simply by subtraction.

There are *p-1* regression degrees of freedom, where *p* is the number of parameters in the graduating function as computed in Equation (5). The number of residual degrees of freedom is simply the number of points in excess of the minimum *p* points required to fit the model, or *n-p*. The degrees of freedom are thus also additive, in that the total df, *n-1*, is just the sum of the *p-1* regression df and the *n-p* residual df. The explained and unexplained variance (*not* additive) are then.

$$\sigma^2_{Regression} = \frac{\sum_{i=1}^{m} r_i \left( \hat{y}_i - \bar{y} \right)^2}{p-1} \tag{13}$$

and

$$\sigma^2_{Residual} = \frac{\sum_{i=1}^{m} \sum_{u=1}^{r_i} \left( y_{iu} - \hat{y}_i \right)^2}{n-p} \tag{14}$$

If there are no replicates, as in a typical computational experiment, the residual variance follows from the general case of Equation (14).

$$\sigma^2_{Residual} = \frac{\sum_{i=1}^{n} \left( y_i - \hat{y}_i \right)^2}{n-p} \tag{15}$$

The residual standard error ("one-sigma value") is simply the square root of Equation (15). No replicates are required to compute this value. If there is no systematic component to the unexplained variance—that is, if the response model represents a good fit to the data—then the residual standard error is a measure of the intrinsic variability in whatever underlying phenomenon is being modeled.

## 4.2 Distribution of Unexplained Variance in a Computational Experiment

The residual variance associated with polynomial graduating functions generated in a computational experiment can be computed by rote application of the defining mathematical relations shown in the previous subsection, simply by summing the squares of residuals and dividing by the number of residual degrees of freedom. However, the *interpretation* of the residual variance depends on the distributional assumptions we are entitled to make. For example, it is colloquially assumed that if precision intervals that are computed from sample statistics have a half-width proportional to "two sigma," they will contain some corresponding population parameter with a probability of 95%. This depends, however, on a number of assumptions about

the unexplained sample variance, including whether it is normally distributed, whether uncertainties in the response estimates are independent, and whether a sufficient number of degrees of freedom are available to estimate the residual variance.

The following anecdotal evidence is offered to suggest that there may not be unanimity of opinion on what distributional characteristics to expect in the residuals of a polynomial regression model fitted to data from a computational code. The author recently put variations of the same question to a half dozen knowledgeable researchers. Three were asked how they would expect the residuals to be distributed about a polynomial fit to data from a computational experiment. Three others were told that residuals from such a computational experiment had been observed to be normally distributed, and were asked to explain how this could occur absent random experimental error.

All three members of the first group forecasted a uniform distribution of residuals, citing their expectation that the unexplained variance would be systematic and therefore non-normal. All three members of the second group claimed to have anticipated a normal distribution, appealing in varying degrees to the Central Limit Theorem. One member of this second group offered evidence in the form of an especially insightful observation [36], noting that colleagues of his with experience fitting graduating functions to data from computational experiments had reported normal distributions of residuals for complex graduating functions involving relatively high-order polynomial functions of multiple variables, but that the residuals were not necessarily normally distributed for simpler models involving low-order polynomial functions in a small number of independent variables.

The author had in fact observed a normal distribution of residuals from the polynomial approximation of a complex computational code, as in Figure 6. This figure displays a normal probability plot for residuals from a reduced third-order polynomial approximation of the complex computational code used to predict thermal reentry loads on the Space Shuttle. Since normally distributed residuals lie along a straight line in such a plot, it is evident that the residuals are indeed Normal, notwithstanding the fact that all response data fitted in this study were computed from completely deterministic computer codes that were apparently devoid of random error. This case study will be discussed in more detail in Section 5.0.

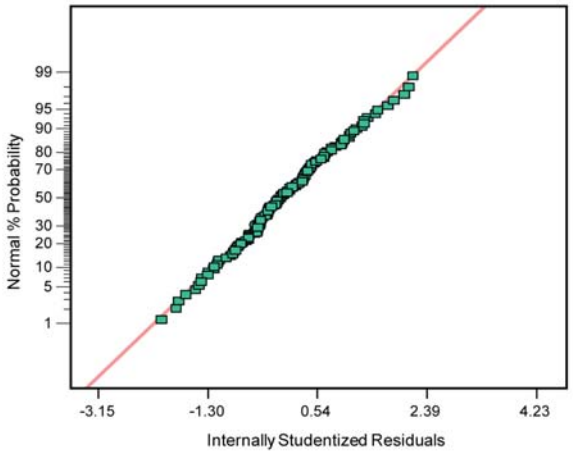


**Figure 6: Normal probability plot of residuals from polynomial approximation of a finite element code.**

Fortunately, the Central Limit Theorem does in fact ensure that prediction errors from well-fitted polynomial response models behave as if they have been drawn from a Normal distribution, as we now explain. Polynomials consist of the algebraic sum of a number of monomials, each of which is comprised of the product of a coefficient and some function of the independent variables. There will be uncertainty in the estimation of each regression coefficient, reflecting inevitable errors in the data used to estimate them. Errors in a low-order polynomial approximation are also introduced when higher-order terms in an infinite Taylor series are dropped. Those truncation errors are shared by the coefficients of the retained terms, except in the special case of orthogonal polynomials [8]**.** The combination of truncation error and ordinary experimental error ensures that each monomial in the graduating function will feature some uncertainty.

The Central Limit Theorem assures us that no matter how the errors in each of the monomials are distributed, the errors in the response estimate that consists of their sum will be *normally* distributed if there are a sufficiently large number of terms, and as long as none of the individual monomial errors dominate the others, conditions that can be expected to hold if the regression fit is satisfactory and the model features a moderate number of independent variables. The more complex the model, the larger the number of monomials in the polynomial graduating function and the more closely the response estimate will approximate a selection from a normal distribution, consistent with the anecdotal evidence offered above.

One component of the uncertainty in polynomial approximations to the underlying code therefore reflects imperfections in the graduating functions and is normally distributed; but it is worth conjecturing whether there might be some component of uncertainty in the underlying code itself that is likewise distributed normally. If, as seems likely, the error in a response estimate from the underlying code can be reasonably regarded as the algebraic sum of multiple errors introduced during the development of the code, then by the Central Limit Theorem errors in the underlying code itself will also behave as if they were drawn from a normal distribution.

Random errors that occur in a physical experiment can be regarded as having a normal distribution when sampled over *time*—the differences between some reference level (such as the sample mean) and the levels of ostensibly identical individual points that are acquired over some interval of time are expected to display a normal distribution. In a computational experiment, the error associated with any given point in the inference space is invariant with time, but such errors can be normally distributed when sampled over *space.* The differences between some reference level (such as a fitted regression model) and the levels of individual points that are acquired at specific sites within some region of the inference space can display a normal distribution, notwithstanding the fact that the error at each site is a completely deterministic bias error. It may seem unusual, especially to a physical experimentalist, to think that components of a *bias* error might be distributed normally. And yet it appears as if at least some components of the error in an ensemble of predictions from a computational code can be simultaneously deterministic *and* normally distributed throughout the inference space.

The residuals associated with well-fitted polynomial approximations to an underlying code simply reflect the difference between an estimate from the graduating function and an estimate from the underlying code. Precision intervals computed from such normally distributed residuals may therefore reflect two sources of uncertainty, the uncertainty associated with a severely truncated polynomial approximation to an underlying code, *and* some component of intrinsic error in the underlying code itself.

Much of the effort in an MDOE experiment is in testing the adequacy of graduating functions that have been developed as an approximation to the true but unknown underlying process (physical or computational). Numerous tests are employed to objectively examine the residuals, looking for the tell-tale signs of significant

systematic error that would indicate a lack of fit. One such test, known as the Lack-of-Fit F-Test can be borrowed from the application of MDOE methods to physical experimentation.

The Lack-of-Fit F-Test requires that genuine replicates be acquired during the execution of an experiment (more on a work-around for computational experiments presently). Replicates are a standard element of MDOE test matrices for this and other reasons, and they allow the residual variance to be further partitioned into lack-of-fit (LOF) and pure error (PE) components. We proceed in the usual way, by computing LOF and PE sums of squares and their corresponding degrees of freedom, as follows:

$$SS_{Pure\,Error} = \sum_{i=1}^{m}\sum_{u=1}^{r_i}\left(y_{iu} - \overline{y}_i\right)^2 \tag{16}$$

and

$$SS_{Lack\,of\,Fit} = \sum_{i=1}^{m} r_i\left(\hat{y}_i - \overline{y}_i\right)^2 \tag{17}$$

As before, these sums of squares are additive; that is, $SS_{Residual} = SS_{Pure\ Error} + SS_{Lack\ of\ Fit}$, so the LOF sum of squares can be computed by subtraction if the residual and pure error sums of squares have already been computed. There are *n-m* pure error degrees of freedom and *m-p* lack-of-fit degrees of freedom, where again *n* is the total number of points in the regression, *m* is the number of unique sites in the inference space, and *p* is the number of parameters in the regression model, including the intercept. As before, these degrees of freedom are additive, so that the LOF and PE degrees of freedom sum to the residual degrees of freedom: *n-p = (n-m) + (m-p)*.

The pure error and lack-of-fit components of the residual variance are computed in the usual way, by dividing the sums of squares by their corresponding degrees of freedom:

$$\sigma_{PE}^2 = \frac{\sum_{i=1}^{m}\sum_{u=1}^{r_i}\left(y_{iu} - \overline{y}_i\right)^2}{n-m} \tag{18}$$

and

$$\sigma_{LOF}^2 = \frac{\sum_{i=1}^{m} r_i\left(\hat{y}_i - \overline{y}_i\right)^2}{m-p} \tag{19}$$

It can be shown that if the pure error is normally distributed, the ratio of the lack-of-fit and pure error variances follow an F distribution. That is,

$$\frac{\sigma_{LOF}^2}{\sigma_{PE}^2} \sim F_{m-p,\,n-m} \tag{20}$$

We can therefore objectively test a null hypothesis of no significant lack of fit against its alternative that the lack of fit is significant, by comparing Equation (20) with tabulated critical F-statistics with *m-p* and *n-m* degrees of freedom. Large and statistically significant F values imply that the model does not fit the data well. This would typically motivate the search for a better model—one with higher-order terms in the polynomial approximation, perhaps, or one that is fitted over a less ambitious range of the independent variables.

One difficulty with the LOF F-test is that when high-precision data are acquired, the denominator in Equation (20) can be so small that the resulting lack-of-fit F-statistic is significant in a statistical sense, but not in a practical sense. That is, lack-of-fit errors may be well within specified tolerances and yet considerably larger than the pure error that characterizes an extremely stable measurement environment. Note that in the limit as pure error approaches zero, this approaches the situation we encounter in a computational experiment.

One solution to the "problem" of high-precision data (a problem only in the sense that it complicates this particular statistical test) may be to test the lack-of-fit component of the residual variance not against the pure error component, but against some specified precision *requirement*. This would obviate the need for an empirical estimate of pure error, and therefore for replicates. Absent the need for replicates, this lack-of-fit test could be applied in a computational experiment as readily as in a physical experiment, to determine if any imperfections in the polynomial graduating functions used to approximate underlying processes are large enough to be of practical concern. If so, we would then seek a more satisfactory graduating function as in the case of a physical experiment, by invoking a higher-order polynomial or constraining the approximation to a reduced range of independent variables.

The LOF F-test described here is one of several tests that can be applied to the residuals of a polynomial graduating function to objectively determine how well a simple polynomial model approximates some complex underlying phenomenon. In a computational experiment, a satisfactory approximation will feature slopes (derivatives) throughout the inference space that closely match those of the underlying code. This facilitates the propagation of uncertainty in the independent variables, as will be demonstrated in the next section.

## 4.3 Error Propagation

The responses predicted by any computational code are necessarily impacted by errors in the specification of independent variable levels, regardless of the quality of the code. In this section, we review how the infinitely differentiable nature of analytic polynomials facilitates the propagation of errors attributable to uncertainty in factor level specifications, when such polynomials are used as graduating function approximations to some complex underlying code.

We use ordinary regression to fit a polynomial model to computational response estimates based on the underlying code. These response estimates are computed for a schedule of independent variable combinations that has been designed by MDOE methods to feature the smallest number of runs adequate for this task. The MDOE design will have optimized the selection of factor level combinations to minimize the effects of uncertainty in the response estimates on polynomial model predictions.

We apply tests of the type described above to determine if the lack-of-fit component of the residual variance associated with such polynomial approximations is insignificant; and if not, we revise the polynomial model by increasing its order, by reducing the range of independent variables over which it is fitted, or perhaps by transforming the response variables or one or more of the independent variables. When we are able to reduce the lack-of-fit component of the residual variance to insignificant levels, we can be reasonably confident that the resulting graduating function parallels the underlying code over the range of independent variables that

have been fitted. This implies that the slopes of the underlying code and the simplified polynomial approximations are similar, which enables the propagation of independent variable errors by means of the polynomial approximating functions.

Consider the following general propagation formula, easily derived from elementary error propagation theory [37], for the case of a function of *k* factors for which errors in their levels are independent of each other:

$$\sigma_y^2 = \left(\frac{\partial y}{\partial x_1}\right)^2 \sigma_{x_1}^2 + \left(\frac{\partial y}{\partial x_2}\right)^2 \sigma_{x_2}^2 + \cdots + \left(\frac{\partial y}{\partial x_k}\right)^2 \sigma_{x_k}^2 \tag{21}$$

Here, $\sigma_y^2$ is the variance in a response, *y*, where that response is a function of *k* independent variables, $x_1$ through $x_k$. The quantities $\sigma_{x_i}^2$ are the error variances for each of the independent variables. The derivatives are evaluated at the values of the $x_i$ for which the response uncertainty is of interest.

There are combinations of the independent variables for which the code is relatively less sensitive to errors in the independent variables, and other combinations for which it is more sensitive. A given level of uncertainty in the independent variables has greater influence where response gradients are relatively steep, and less influence where the response function is relatively independent of the factor levels.

Except for the uncertainty in each of the individual independent variables, the propagated response uncertainty depends only on the first derivatives of the underlying response function with respect to each independent variable. As noted above, because of explicit tests from the MDOE analysis to ensure that lack of fit is insignificant in the polynomial graduating functions used to represent the underlying computational codes, we are justified in assuming that the polynomial approximating functions adequately match the underlying code so that their derivatives are numerically similar. These derivatives can be difficult and/or costly to obtain from the underlying codes directly, but they are trivial to determine from the infinitely differentiable polynomial graduating functions. We therefore substitute the derivatives from the polynomial graduating functions for the derivatives in the underlying code. Equation (21) then facilitates a straightforward estimation of response errors due to errors in the independent variable levels.

## 5.0 SUMMARY OF A CASE STUDY: PROPAGATION OF UNCERTAINTY IN SPACE SHUTTLE REENTRY LOADS

This section illustrates the methods outlined in this paper by summarizing results recently published in a paper titled “Space Shuttle Debris Impact Tool Assessment Using the Modern Design of Experiments” that was presented at the 45th AIAA Aerospace Sciences Meeting and Exhibit in Reno, NV, USA in January 2007. That paper was a collaboration among the present author and four colleagues at Johnson Space Center: Elonsio M. Rayos, Charles H. Campbell, Steven L. Rickman, and Curtis E. Larsen. Techniques described in the current paper were used in that collaboration to develop compact, resource-minimal MDOE computational experiments that could produce polynomial approximations to the underlying codes used to predict thermal and structural reentry loads on the Space Shuttle that are associated with blemishes in the Shuttle’s Thermal Protection System (TPS). These blemishes are induced by debris impact—typically ice or foam—on ascent.

The windward side of the Shuttle orbiter is divided into several regions called Body point Zones (BPZ). Figure 7 illustrates these zones, which are distributed symmetrically about the longitudinal centerline. The applied thermal and structural loads differ from one BPZ to another on reentry, the thermal protection tiles differ in thickness, and the underlying structural composition of the orbiter is also different from one zone to another. For these reasons, a separate MDOE computational experiment was designed for each of the 33 zones.

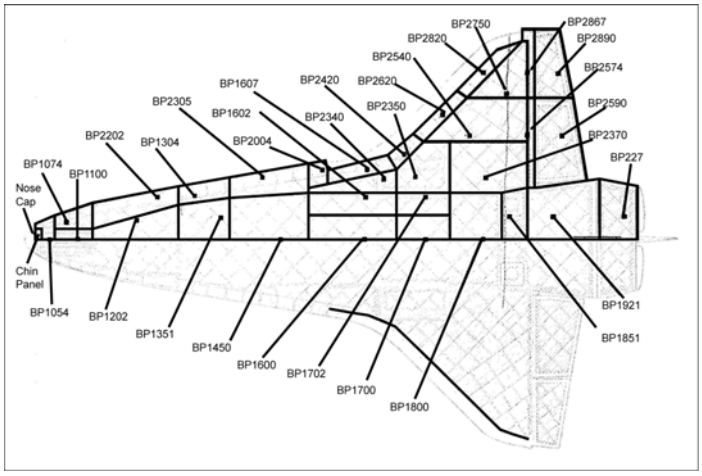


**Figure 7: Body Point Zones on Shuttle Orbiter.**

There were seven independent variables in each experiment. Five of these were numerical cavity geometry variables: length, width, depth, entry angle, and side angle. There were also two categorical variables—variables that can take on discrete values only, rather than arbitrary values within a prescribed range. These were baseline heating ("low" and "nominal"), and two prescribed values of boundary layer transition time (surrogates for surface roughness). As Equation (5) indicates, the parameter count for a response surface model, and therefore the number of runs required in a computational experiment to develop such a model, increases rapidly with the number of independent variables.

We mention in passing that the originally proposed experiment featured *ten* independent variables rather than seven. An especially compact MDOE experiment design was applied in what is known as a screening experiment to reveal that three of the originally proposed variables had relatively little effect on the response estimates of interest, and could be eliminated from the study with negligible impact. This resulted in a reduction from 284 runs per zone that would have been required to support a ten-variable experiment, to 124 runs, a savings of 56% of the required CPU time per zone. This resulted in a savings of 5280 runs of the underlying numerical codes across all zones, or roughly 500 hours of expensive CPU time.

After discussions with subject-matter experts at Johnson Space Center, it was decided that for the response variables of interest, third-order polynomial models would adequately approximate the underlying computational codes over the ranges of independent variables to be considered. Equation (5) can be used to compute the number of parameters in a *full* $d^{th}$-order model in $k$ variables. But because two of the independent variables were two-level categoricals for which pure quadratic and higher-order terms did not have to be estimated, a slight modification to Equation (5) revealed that 104 total terms were required for the full model, meaning that each MDOE computational experiment would require a minimum of 104 runs.

In the MDOE experiment designs as executed, another 20 lack-of-fit degrees of freedom were added to accommodate terms of higher order than three that might have to be added to drive LOF errors to negligible levels. So responses were fit to a total of 124 points. In addition, another 15 "confirmation points" were acquired in each zone. These are points that are not used to fit the response models but are held in reserve to test those models. They consist of randomly selected levels of the independent variables within the ranges examined. The polynomial models developed from the 124 regression points are used to predict responses for each of the 15 confirmation points. These predictions are compared with response estimates developed from the full underlying code. A Critical Binomial Analysis is performed to determine if a minimum number of successful confirmations is achieved. If not, the polynomial approximating model is modified as described in the previous sections. Including the 15 confirmation points, the minimum 104 regression points, and the 20 points added arbitrarily to allow corrections for lack of fit, there were a total of 139 runs of the underlying code specified for the MDOE computational experiment that was executed in each Body Point Zone.

There were six thermal and structural response variables of interest. (There is no limit in principle to the number of response variables that can be modeled with a given MDOE design, and the test matrix is not affected by how many responses are to be recorded for each combination of independent variables.) A total of 198 polynomial response models were therefore generated, six for each of the 33 Body Point Zones.

The original estimate of third order responses proved to be more than adequate; excellent fits were achieved (negligible lack-of-fit components in the unexplained variance) with significantly reduced third-order models that featured typically a quarter to a third of the total number of terms provided for in the design. The final polynomial response models normally included significant first-order (linear) terms for all independent variables, numerous factor-interaction terms, and pure quadratic terms for some subset of the numerical independent variables (cavity geometry factors). There were typically a few mixed cubic terms, but pure cubic monomials were rarely needed to achieve a good fit.

Residuals were generated by subtracting the response estimates produced by the full underlying code from the corresponding estimates developed from the polynomial approximating functions. These residuals were examined in a number of ways to test the polynomial models for lack of fit, as described in part in the previous sections. Figure 8 is a representative display of residuals—for structure temperature in BPZ 1602 in this case.

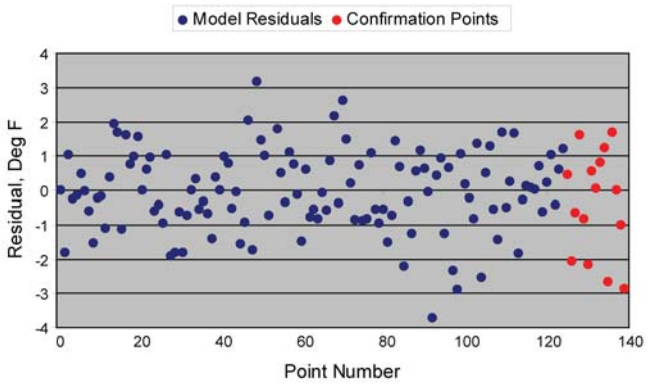


**Figure 8: Example of residuals: Structure Temperature for BPZ 1602. Standard deviation: 1.2° F. Residuals at the 15 confirmation points (red) are comparable to the 124 regression residuals (blue).**

Several observations can be made in this figure. First, the residuals display no pattern when plotted as a function of point number, a surrogate for time. This is less noteworthy in a computational experiment than in a physical experiment, but it is reassuring nonetheless. It implies that there were no significant time-varying systematic effects biasing the response estimates while the experiment was being executed. In a physical experiment, temperature changes, operator learning and fatigue effects, drift in the instrumentation and data systems, and an unknown (and unknowable) number of other persisting (non-random) effects can all conspire to generate time-varying bias errors in a physical experiment.

In a computational experiment such as this one, certain changes could have been made in the middle of the experiment that bias the post-change results relative to the pre-change results, and we would expect this to be revealed through some shift in the residuals where the change occurs. For example, operators might decide in the middle of an experiment that certain efficiencies can be achieved in the execution of the code (i.e., it might be made to run faster) if grid geometry specifications were altered slightly, or if some subroutine thought to be contributing relatively little were turned off to save overall execution time. Such mid-test alterations to the "test article" (in this case, the underlying code) are discouraged in MDOE testing, but they are not unheard of and in fact are surprisingly common. An examination of the residuals helps identify these events.

The second noteworthy observation that can be made from the structure temperature residuals displayed in Figure 8 is that their standard deviation is only 1.2° F, which represents a coefficient of variation in this case of only 0.48%. The original goal of this polynomial approximation activity was to estimate thermal response predictions made by the underlying code within 10° F. Clearly, the polynomial approximations are well within this tolerance. This level of agreement was observed generally across all 33 Shuttle Body point Zones.

Finally, it is interesting to compare the 15 confirmation-point residuals displayed in red in Figure 8 with the 124 regression residuals that are displayed in blue. Note that in terms of their magnitude and distribution, the confirmation residuals are indistinguishable from the regression residuals except by their color. This suggests that the response model is as effective at estimating responses for independent variable combinations that were *not* included in the development of the model, as it is for points that were included in the regression.

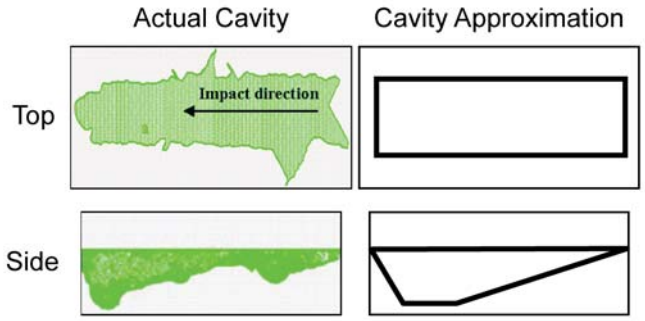


**Figure 9: Uncertainty in specification of impact cavity dimensions is introduced by simplified geometry required by the underlying code.**

Having assured ourselves by examining the residuals in a number of ways that the polynomial models approximated the full underlying code adequately, we turned to the problem of error propagation. Figure 9 illustrates the need for error propagation in this application. The thermal and structural response variables were all functions of the independent variables noted above, which included five cavity geometry variables. Unfortunately, the underlying numerical code cannot accommodate a full description of the complex

geometry that characterizes a typical debris strike. Instead, an idealized "shoebox" geometry is used as an approximation. Each shoebox can be described only in terms of its length, width, and depth, the angle of its two sides (assumed to be constant and equal), an entry angle and an exit angle. (The exit angle was eliminated as a significant variable in the screening process described above.)

As Figure 9 illustrates, the necessarily idealized cavity description results in some uncertainty in the specification of cavity geometry. It was of considerable interest to be able to quantify how this uncertainty affected the estimates of thermal and structural reentry loads on the Shuttle. The polynomial approximations to the underlying numerical codes facilitated this task quite easily. Response models were easily generated from the polynomial approximating functions, and from their derivatives and specified levels of uncertainty in the geometry variables, models for response uncertainty could also be developed. Figure 10 displays structure temperature as a function of two normalized cavity geometry variables for BPZ 2540. The uncertainty in structure temperature estimates due to specified cavity geometry uncertainty is also displayed for the same geometry variables. It is clear that both the structural temperature and the contributions that cavity geometry errors make to the uncertainty in estimating it are greater for longer, deeper cavities than for shorter, shallower cavities, a result that is not inconsistent with intuitive expectations.

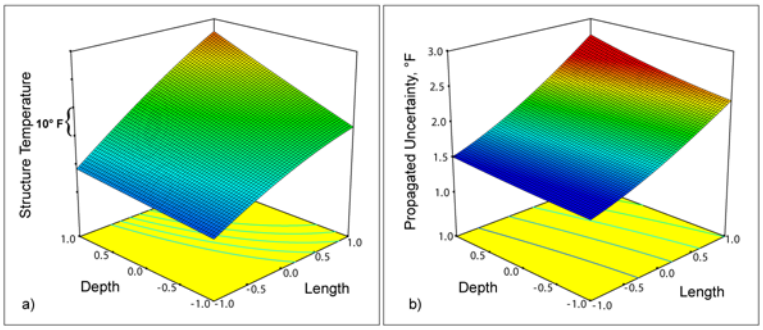


**Figure 10: Structure Temperature for representative cavity in BPZ 2540. a) Response surface for temperature, b) Response surface for uncertainty in temperature due to uncertainty in cavity geometry variables.**

This section has reported a small subset of published results obtained by applying MDOE experiment design methods to develop low-cost, high-quality surrogate models to approximate the complex codes used to estimate Shuttle reentry loads. The interested reader can find more details in references [31] and [32]. The intent here has been simply to use a practical application to illustrate some of the methods described in previous sections of this paper.

## 6.0 DISCUSSION

It is important to recognize that in a computational experiment in which response estimates for specified independent variable combinations are computed from an underlying code, it is the underlying code that represents truth to the experimentalist. This means that polynomial graduating functions developed in such experiments will only approximate the *underling code*, not necessarily the physical phenomenon that the underlying code purports to describe. An inherent assumption is that the code is has been *validated*, so that its response estimates are an adequate approximation of what occurs in nature.

The Modern Design of Experiments can be used to design very compact, resource-minimal validation experiments by which the underlying code is compared with physical measurements. In such experiments the

response of interest is a physical/computational differential estimate quantifying the validation errors in the underlying code as a function of independent variables. This information can be used to support an objective certification of the code. It can also be used to identify regions in the inference space where the code generates less than reliable response estimates, with a view to improving the code in those regions. Such results may provide insights into specific weaknesses of the code that can be addressed, or they may identify inference space boundaries within which the underlying code can be used with confidence.

Non-zero validation errors are associated with any model developed with finite resources, even if the underlying code is believed to be adequately validated in some overall sense. That is, responses estimated at each site in the inference space will be imperfect, even if they are generally adequate for a particular purpose. They will suffer some non-random error that systematically biases the response estimate at that point somewhat high or somewhat low compared to the true response.

We have invoked the Central Limit Theorem to suggest that at least some components of these errors are normally distributed so that an adequately fitted regression model will display normally distributed residuals, and we have cited evidence of such a distribution from the case study outlined in the previous section. This suggests that precision intervals developed from the residuals of well-fitted polynomial graduating functions will approximate the uncertainty in predictions from the underlying computational code. However, there will also be some intrinsic error in the polynomial graduating function itself, caused by the truncation of higher-order terms (unless the polynomial is orthogonal, which is a special case). These errors are also expected to be normally distributed by the Central Limit Theorem, as noted previously. Residuals from a polynomial graduating function can therefore be expected to be attributable to two sources of error: finite-polynomial truncation errors and validation bias errors that are not systematically distributed throughout the inference space.

Note, however, that these residuals only represent that component of the total error that can be revealed by fitting computational response estimates to a regression model. There may be additional components of the total error that are attributable to validation errors that either are not a function of the independent variables, or are distributed as some systematic function of the independent variables. Such “stealth” errors would not be apparent in the residuals from a computational experiment, but they would still contribute to an overall bias error in the underlying code.

A particularly insidious version of this type of systematic error can occur in physical experiments in which the set-points of an independent variable are changed monotonically with time, as when the angle of attack in a wind tunnel test is varied in uniform intervals from some relatively low level to some relatively high level, with each new point at a higher angle of attack. If an unknown covariate effect is generating systematic response changes at the same time (frictional heating that might cause an expansion of the test section that impacts wall corrections, for example), then the apparent relationship between measured responses (forces and moments, say) and some independent variable (say, angle of attack) will actually reflect the sum of the true effect and the covariate effect. There will be no way to detect that such an error has even occurred, much less to quantify it. The errors that are attributable to systematically changing covariate effects in a physical experiment generally become apparent only much later, and then only if the experiment is repeated under conditions for which either no covariate effects are in play, or covariate effects are in play that are different from the first experiment.

Randomization of the set-point order defends against systematically changing covariate effects, and is a standard MDOE quality assurance tactic in physical experiments. Randomizing the set-point order ensures that any given level of the independent variable has an equal chance of being acquired early or late, so there is an equal probability that any covariate effects that are in play will generate systematic errors that are positive or negative relative to the sample mean. That is, randomizing the set-point order has the effect of converting otherwise undetectable systematic errors into just another component of random error—easy to detect and easy to control by replication. The shape of the response model—its slopes at each point in the inference space—remains unchanged by the covariate effects. Only the intercept term of the model is affected, by an amount that can also be quantified as an additional component of bias error. (Quantifying this important component of bias error requires that the experiment design be "blocked" appropriately, a topic that is beyond the scope of the present paper but which is also a standard MDOE quality assurance tactic in physical experiments, and which is addressed elsewhere in the literature [3,4,7]).

It is unfortunate that in a computational experiment, any bias errors in the underlying code that may be a function of the independent variables will not be influenced by the order in which the points are computed. Set-point randomization therefore is not a viable quality assurance tactic in a computational experiment as it is in a physical experiment. Furthermore, it is possible that there are systematic errors that are not even a function of the independent variables at all (the classic "eleven-inch ruler" problem that results in constant bias errors throughout the inference space).

To the extent that such undetectable components of bias error may be present, we cannot use the precision intervals computed from residuals in a computational experiment as a reliable estimate of the total uncertainty in the underlying code. On the other hand, given the accommodatingly normal distribution of such residuals, those precision intervals can be expected to constrain the location of response estimates made by the underlying code, and therefore serve to indicate how well the polynomial graduating functions approximate the code, independent of how well the code predicts reality. Assuming a good fit of the computational response estimates to the polynomial graduating function (no significant systematic error component as revealed by normal probability plots and other tests of the residuals), the uncertainty derived from such residuals will be due to polynomial truncation errors (a measure of the imperfection of that approximating function), and to those bias error components in the underlying code that are not systematically distributed throughout the inference space.

The well-behaved nature of polynomial response residuals ensures that properly-fitted polynomial approximations can be made to approach the underlying code arbitrarily closely simply by increasing the volume of data acquired in the computational experiments that generate them. The fidelity with which the polynomial graduating functions can be made to approximate the underlying code suggests a certain synergism that can be achieved by the merger of low-cost formal experiment design methods with conventional computational techniques. One can easily envision a process in which a relatively few runs of a complex and expensive underlying code are exercised to generate response estimates for a compact, resource-minimal computational MDOE experiment. Graduating functions developed from the MDOE experiment are tuned to eliminate significant lack of fit. Such functions can then be executed with little cost and CPU time as a surrogate for the underlying code to provide reasonable response approximations. This could be especially advantageous in high-volume applications such as the creation of comprehensive databases, where execution costs could be prohibitive if the full underlying code were to be used for every point, but where the corresponding polynomial approximations could be invoked for negligible cost. The underlying code can always be executed to check any questionable response estimates made by the polynomial approximating

functions (an extension of the basic confirmation-point quality assurance tactics described above, which are already incorporated in the MDOE process).

Under the proposed scenario, the full, complex, and costly underlying computational code would fulfill the critical role of providing high-quality response estimates for each combination of independent variables in a compact MDOE test matrix. The polynomial graduating functions that result from the MDOE experiment could then be used for high-volume database construction.

We close this discussion section with a few remarks about a different topic. In order to use low-order polynomials as a valid approximation for some underlying code, it is frequently necessary to develop separate polynomial models in different regions of the complete inference space. This is almost always the case when the inference space spans regions in which the underlying physics changes across the boundary, as happens for example when we develop different models for aircraft forces and moments in the pre-stall, stall, and post-stall angle of attack regions. Likewise, we would typically establish inference subspace boundaries to separate Mach ranges into subsonic regions of compressible and incompressible flow, as well as regions of transonic and supersonic flow. It would not be wise to fit so many different physical phenomena with a single low-order polynomial approximation.

This inference subspace truncation results in a piecewise continuous response surface spanning the entire inference space that resembles a patchwork quilt of individual models. Inevitably, there are boundary discontinuities between the response surfaces of adjacent inference subspaces. That is, the levels and slopes of adjacent response surfaces do not generally match for every point (or typically for any point) on the boundary.

For those researchers whose analysis procedures rely heavily on graphical displays of data, such inference subspace boundary discontinuities can be unsettling. However, there is no reason to feel uneasy about this. The “discontinuity” at the inference subspace boundary simply reflects the fact that boundary response estimates are made from two different models, each with some prediction uncertainty. It would be just as coincidental if two such models predicted the same response at the boundary as it would be if two replicated measurements made for the same combination of independent variables yielded identical results in a physical experiment. In truth, there is prediction uncertainty throughout each of the subspaces. The boundaries are unique only insofar as the uncertainty becomes evident there. There are generally no physical discontinuities at the inference subspace boundaries, which are, after all, selected more or less arbitrarily by the researcher. It is more accurate to say that there might be boundary discontinuities in the response surfaces representing the upper and lower limits of some precision interval that locates the true response, but the true response surface itself is continuous across the boundary and lies between the precision interval limits on both sides of the boundary.

## 7.0 CONCLUDING REMARKS

Physical and computational experiments share many common elements. Both seek to quantify relationships between selected response variables of a system under study and the levels (settings) of independent variables that influence those responses. The defining distinction is largely inconsequential; namely, that in a physical experiment direct measurements supply the information on system response while in a computational experiment responses are calculated with a computer code.

Certain other practical distinctions appear relevant, however, especially in the context of estimating uncertainty in either class of experiment. Chief among these is that there is typically no error variance in a sample of computational replicates acquired over time at a fixed site in the inference space (unless error components of the response have been explicitly modeled). However, assuming only that the responses vary with independent variable levels, there is always variance in a sample of computational responses estimated over *multiple* sites within the inference space. That variance can be partitioned into explained and unexplained components by methods reviewed in this paper. The unexplained component of the total variance so computed can be used to quantify at least part of the uncertainty in the underlying code.

Significant efficiencies in the design, execution, and analysis of physical experiments have been achieved at Langley Research Center through the application of a formal experimental process known at Langley as the Modern Design of Experiments (MDOE). The objective of experimentation from an MDOE perspective is to extend the researcher's ability to adequately forecast future system behavior; we abandon the traditional notion that data acquisition per se is the primary objective of experimentation. This fundamental change in the perception of experimentation and its purpose influences how concepts of quality and productivity are perceived. Traditional data-centric metrics ("data quality," "data volume") are replaced with a knowledge-management focus on making reliable inferences at low cost. This has resulted in significant reductions in cycle time, operating costs, and uncertainty in a broad range of physical experiments, as documented in the references.

The practical improvements in quality and productivity that have been achieved by applying MDOE methods to physical experimentation, coupled with a realization that there are more similarities between computational and physical experimentation than there are differences, has fostered interest in applying MDOE techniques to computational experimentation, with a view to reducing both cost and uncertainty. This paper has described an application of the Modern Design of Experiments (MDOE) to produce compact, resource-minimal test matrices for computational experiments. The results of these MDOE experiments are analyzed to generate polynomial functions of the independent variables that serve as a surrogate for the more complex underlying code. The cost of executing the polynomial response models is generally much less than the cost of executing the underlying code, which is especially important in applications for which many case runs are required, as when a large database is to be populated. Quality *assurance* tactics have been described by which lack-of-fit errors are minimized by site selection in the inference space, and quality *assessment* techniques have been described by which significant lack-of-fit errors that may exist are detected. Such lack of fit signifies the need to improve the surrogate polynomial models. Common quality improvement strategies have been outlined.

Means for dealing with certain practical complications were discussed, including the reality of dealing with complex response functions over wide inference spaces. Unanticipated distributional properties of prediction residuals were also presented and discussed.

The special utility of polynomial response models for propagating uncertainty in independent variable settings was described and illustrated with a specific case study. The case study dealt with a recent assessment of uncertainty in computational estimates of Space Shuttle reentry loads attributable to ice and foam debris impact on ascent.

Key points from this paper are summarized as follows:

1. Computational codes developed to estimate the response of complex systems to a specified set of conditions provide system response estimates that can be costly and time-consuming to generate, and it is inherently difficult to assess the uncertainty of such estimates.

2. Complex system responses can be adequately represented over suitably constrained independent variable ranges by a Taylor series that is truncated to a simple low-order polynomial in the independent variables, known as a graduation function.
3. Such graduating functions can be developed from a minimum number of strategically selected case runs of the underlying code by invoking the Modern Design of Experiments (MDOE), a low-cost method of experimentation with minimal data volume requirements and integrated quality assurance and quality assessment procedures.
4. The Modern Design of Experiments was originally developed to improve quality and productivity in physical experiments and has been applied for many years at Langley Research Center in wind tunnel testing and other applications, but it can be applied in computational experiments as well as physical experiments.
5. The adequacy with which a low-order polynomial graduating function represents the more complex underlying computational code is revealed through an examination of residuals representing the difference between response estimates made with the underlying code and its polynomial approximation.
6. The residual variance in a computational experiment can be quantified, and objectively tested against specified uncertainty tolerance levels to determine if it is large enough to justify expending the resources required to improve the model.
7. A better fit can be achieved between a polynomial approximating function and the underlying computational code by increasing the order of the polynomial or by altering the range of independent variables over which the polynomial is fitted.
8. Residuals from computational experiments that generate well-fitted graduating functions are normally distributed.
9. The well-behaved distributional properties of residuals from a computational experiment can be used to generate meaningful precision intervals.
10. Precision intervals from a computational experiment quantify the uncertainty due to two sources of error—truncation errors induced by retaining only low-order terms to represent an infinite Taylor series, and components of the bias errors in the underlying code that do not vary systematically with the independent variables, and which are normally distributed by the Central Limit Theorem.
11. Precision intervals from a computational experiment cannot account for errors that may be a systematic function of the independent variables, or classical bias errors that are constant for all combinations of independent variable levels.
12. If the underlying computational code has been adequately validated, we may assume that bias errors are insignificant, including both constant bias errors and bias errors that may be a function of the independent variables.
13. Precision intervals developed from the residuals of a computational MDOE experiment will constrain the location of the underlying code.
14. For an adequately validated computational code, precision intervals developed from the residuals of an MDOE computational experiment that are used in conjunction with the surrogate graduating function can be expected to constrain the location of the true physical response with a specified frequency. That is, response estimates made with the underlying code will lie within a precision interval half-width of response estimates made with the polynomial graduating function, and if the underlying code faithfully predicts the physical phenomenon, the true response will likewise fall

within the precision interval centered on the response estimate generated by the polynomial response model.

15. Polynomial approximations to a computational code can provide insights into the underlying system that are difficult to obtain with a complex computer code. These include indications of the relative sensitivity of system responses to each independent variable, and also information about how the responses of interest are influenced by interactions among the independent variables.
16. A well-fitted polynomial approximation to an underlying code will match the underlying code in level and slope throughout the region of the inference space for which it is valid. That is, absent systematic lack-of-fit error, the numerical values of the first derivatives of polynomial approximating functions can be expected to match those of the underlying numerical codes.
17. When the first derivatives of a polynomial graduating function match those of the underlying computational code, errors in specifying the independent variables can be easily propagated into a potentially important component of the total uncertainty in the underlying code.
18. Apparent discontinuities in polynomial response models across the boundary between two inference subspaces are no cause for concern, as they simply reflect ordinary model prediction uncertainty that exists throughout the inference space, and not just on the boundaries.
19. MDOE methods have been applied in numerous practical computational experiments to develop surrogate polynomial approximations to the complex underlying codes used to forecast thermal and structural loads experienced by the Space Shuttle on reentry, due to debris strikes on the windward side of the orbiter on ascent.
20. Preliminary, low-cost MDOE screening experiments reduced the number of candidate factors in the Shuttle computational experiment from ten to seven. This resulted in a reduction of 5,445 computer runs with an estimated savings of over 500 hours of CPU time.
21. Relatively low-order polynomials in seven independent variables were capable of adequately representing Shuttle reentry codes, with most responses well-fitted with reduced quadratic models augmented with a small number of third-order terms.
22. Residuals from the Shuttle debris impact experiment were observed to have a normal distribution.
23. The widths of the 95% precision intervals developed from the normal distributions of residuals from Shuttle computational experiments were well within specified precision goals for all Shuttle reentry response variables, with MDOE models predicting structure temperature within 2% of the underlying numerical codes.
24. Confirmation points were found to consistently fall within prediction intervals based on residuals from the Shuttle computational experiments that were executed to develop polynomial approximations to the underlying code.
25. Significant resource savings can be achieved by invoking the polynomial response surface modeling techniques of an MDOE computational experiment to predict the underlying code's response estimates throughout the inference space, rather than relying on individual runs of the underlying code for every unique independent variable combination of interest. The following general process is proposed as a way to exploit techniques from the Modern Design of Experiments to improve productivity and quality:
    - Develop the underlying code
    - Use that code to estimate system responses for an MDOE test matrix

- Fit the computational responses to low-order polynomials over suitably truncated ranges of the independent variables
- Validate the fidelity with which the low-order polynomials approximate the complex underlying code
  - Examination of residuals for tell-tale signs of systematic error
  - Confirmation points
- Use the polynomial response functions as low-cost, high-speed surrogates for the underlying code
- Estimate general uncertainty in the response predictions by partitioning the total variance in the MDOE experiment into explained and unexplained components
- Use the polynomial surrogate functions to propagate errors in the independent variables into uncertainty in the response estimates

## 8.0 REFERENCES


[1] DeLoach, R. (1998). *Applications of Modern Experiment Design to Wind Tunnel Testing at NASA Langley Research Center.* AIAA 98-0713. 36th Aerospace Sciences Meeting and Exhibit, Reno, NV, USA.

[2] DeLoach, R. (1998). *Tailoring Wind Tunnel Data Volume Requirements through the Formal Design of Experiments.* AIAA 98-2884, 20th AIAA Advanced Measurement and Ground Testing Technology Conference, Albuquerque, NM, USA.

[3] DeLoach, R. (2000). *Improved Quality in Aerospace Testing Through the Modern Design of Experiments (Invited)* AIAA 2000-0825. 38th AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, USA.

[4] DeLoach, R. (2002). *Tactical Defenses Against Systematic Variation in Wind Tunnel Testing*. AIAA 2002-0885. 40th AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, USA.

[5] Erickson, G.E., Burner, A.W., & DeLoach, R. (1997). *Pressure-Sensitive Paint and Video Model Deformation Systems at the NASA Langley Unitary Plan Wind Tunnel.* High-Speed Research Aerodynamic Performance Workshop, Hampton, VA, USA.

[6] Ciancarelli, C.R., & Dorsett, K.M. (2000). *Optimizing the F-16 Conformal Fuel Tank Using Design of Experiments*. AIAA 2000-4522. 18th AIAA Applied Aerodynamics Conference, Denver, CO, USA.

[7] DeLoach, R., Hill, J.S., & Tomek, W.G. (2001). *Practical Applications of Blocking and Randomization in a Test in the National Transonic Facility (Invited).* AIAA 2001-0167. 39th AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, USA.

[8] Morelli, E.A., & DeLoach, R. (2001). *Response Surface Modeling Using Multivariate Orthogonal Functions (Invited)*. AIAA 2001-0168. 39th AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, USA.

[9] DeLoach, R. (2002). *Applications of the Modern Design of Experiments at NASA Langley Research Center (Invited).* Proceedings of the American Statistical Association, Section on Physical and Engineering Sciences [CD-ROM]: New York, NY, USA.

[10] Morelli, E.A., & DeLoach, R. (2003). *Ground Testing Results Using Modern Experiment Design and Multivariate Orthogonal Functions (Invited).* AIAA 2003-0653. 41st AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, USA.

[11] DeLoach, R., & Erickson, G.E. (2003). *Low-Order Response Surface Modeling of Wind Tunnel Data Over Truncated Inference Subspaces*. AIAA 2003-0456. 41st AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, USA.

[12] Luner, J., & Healey, M. (2003). *Modern Design of Experiments Techniques to Optimize a Leading Edge Extension Fence (Invited)*. AIAA 2003-0655. 41st Aerospace Sciences Meeting and Exhibit, Reno, NV, USA.

[13] Healey, M. (2003). *F/A-18 E/F Vertical Tail Buffet, Design, Analysis and Test*. AIAA 2003-1886. 44th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Norfolk, VA, USA.

[14] Bacon, B.J., DeLoach, R., Gregory, I.M., & Washburn, A. (2003). *Modern Design of Experiments Approach to Modeling Vehicle Force and Moment Main Effects and Interactions for an Advanced UAV with Synthetic Jets*. AIAA Guidance, Navigation, and Control Conference and Exhibit, Austin, TX, USA.

[15] DeLoach, R., & Berrier, B.L. (2004). *Productivity and Quality Enhancements in a Configuration Aerodynamics Test Using the Modern Design of Experiments*. AIAA 2004-1145. 42nd AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, USA.

[16] Dowgwillo, R.M., & DeLoach, R. (2004). *Using Modern Design of Experiments to Create a Surface Pressure Database From a Low Speed Wind Tunnel Test*. AIAA 2004-2200. 24th AIAA Aerodynamic Measurement Technology and Ground Testing Conference, Portland, OR, USA.

[17] DeLoach, R. (2006). *The Modern Design of Experiments for Configuration Aerodynamics: A Case Study*. 44th AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, USA.

[18] Albertani, R., DeLoach, R., Stanford, B., Hubner, J.P., & Ifju, P.S. (2006). *Wind Tunnel Data Base Development and Nonlinear Modeling Applied to Powered Micro Air Vehicles with Flexible Wing (Invited)*. AIAA 2006-6640. AIAA Atmospheric Flight Mechanics Conference and Exhibit, Keystone, CO, USA.

[19] DeLoach, R. (2000). *A Factorial Data-Rate and Dwell-Time Experiment in the National Transonic Facility*. AIAA 2000-0828. 38th AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, USA.

[20] Underwood, P., Everhart, J., & DeLoach, R. (2001). *National Transonic Facility Wall Pressure Calibration Using Modern Design Of Experiments (Invited).* AIAA 2001-0171. 39th AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, USA.

[21] Parker, P., & DeLoach, R. (2001). *Response Surface Methods for Force Balance Calibration Modeling*. 19th International Congress on Instrumentation in Aerospace Simulation Facilities, Cleveland, OH, USA.

[22] Parker, P., & DeLoach, R. (2002). *Structural Optimization of a Force Balance using a Computational Experiment Design (Invited)*. AIAA 2002-0540. 40th AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, USA.

[23] DeLoach, R., Cler, D., & Graham, B. (2002). *Fractional Factorial Experiment Designs to Minimize Configuration Changes in Wind Tunnel Testing*. AIAA 2002-0746. 40th AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, USA.

[24] Cutler, A., Danehy, P., Springer, R., DeLoach, R., & Capriotti, D.P. (2002). *CARS Thermometry in a Supersonic Combustor for CFD Code Validation*. AIAA 2002-0743. 40th AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, USA.

[25] Burner, A.W., Liu, T., & DeLoach, R. (2002). *Uncertainty of Videogrammetric Techniques used for Aerodynamic Testing*. AIAA 2002-2794. 22nd AIAA Aerodynamic Measurement Technology and Ground Testing Conference, St. Louis, MO, USA.

[26] Danehy, P.M., Dorrington, A., Cutler, A.D., & DeLoach, R. (2003). *Response Surface Methods for Spatially-Resolved Optical Measurement Techniques (Invited)*. AIAA 2003-0648. 41st AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, USA.

[27] DeLoach, R. (2003). *Putting Ten Pounds in a Five-Pound Sack: Configuration Testing with MDOE*. 21st AIAA Applied Aerodynamics Conference, Orlando, FL, USA.

[28] Cutler, A.D., Danehy, P.M., Springer, R.R., O'Byrne, S, Capriotti, D.P., & DeLoach, R. (2003). *Coherent Anti-Stokes Raman Spectroscopic Thermometry in a Supersonic Combustor*. AIAA Journal, Vol. 41, No. 12, pp. 2451–2459, December 2003.

[29] DeLoach, R., & Rhode, M.N. (2005). *Short-Duration, High-Quality Wind Tunnel Calibration*. 1st Joint Meeting of the Supersonic Tunnel Association International and the Subsonic Aerodynamic Testing Association, Buffalo, NY, USA.

[30] Rhode, M.N., & DeLoach, R. (2005). *Hypersonic Wind Tunnel Calibration Using the Modern Design of Experiments*. 41st AIAA/ASME/SAE/ASEE Joint Propulsion Conference and Exhibit, Tucson, AZ, USA.

[31] DeLoach, R., et al. (2007). *Space Shuttle Debris Impact Tool Assessment Using the Modern Design of Experiments*. 45th AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, USA.

[32] Yuan, J., & Waguespack, G. (2006). *Orbiter Tile Damage Analysis Tools: Analysis & Assessment*, Structural Analysis Section, Engineering and Science Contract Group, Jacobs Technology, Houston, TX, USA.

[33] Box, G.E.P., & Draper, N. (1987). *Empirical Model-Building and Response Surface*. New York: John Wiley and Sons.

[34] Myers, R.H., & Montgomery, D.C. (2002). *Response Surface Methodology: Process and Product Optimization Using Designed Experiments*, Wiley Series in Probability and Statistics, 2nd ed., New York: John Wiley and Sons.

[35] Giunta, A.A., et al. (2003). *Overview of Modern Design of Experiments Methods for Computational Simulations*. AIAA 2003-0649. 41st AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, USA.

[36] Peixoto, Julio L. The Boeing Company. Private communication.

[37] Coleman, H.W., & Steele, W.G. (1989). *Experimentation and Uncertainty Analysis for Engineers*. New York: John Wiley and Sons.