INSTITUTE FOR DEFENSE ANALYSES

Space Filling Designs for Modeling & Simulation Validation

Heather Wojton, Project Leader

Kelly Avery
Han Yi
Curtis Miller

June 2021

IDA Document NS D-21562

Log: H 2021-000048

INSTITUTE FOR DEFENSE ANALYSES
4850 Mark Center Drive
Alexandria, Virginia 22311-1882

The Institute for Defense Analyses is a nonprofit corporation that operates three Federally Funded Research and Development Centers. Its mission is to answer the most challenging U.S. security and science policy questions with objective analysis, leveraging extraordinary scientific, technical, and analytic expertise.

About This Publication
This work was conducted by the Institute for Defense Analyses (IDA) under contract HQ0034-19-D-0001, Task 229990 "Test Science Applications," for the Office of the Director, Operational Test and Evaluation. The views, opinions, and findings should not be construed as representing the official position of either the Department of Defense or the sponsoring organization.

Acknowledgments
The IDA Technical Review Committee was chaired by Mr. Robert R. Soule and consisted of Christopher Pellegrinelli, Jason Sheldon, James Rhoads, and Sabrina Lyn Hiner Dimassimo from the Operational Evaluation Division.

For more information:
Heather Wojton, Project Leader
hwojton@ida.org • 703-845-6811

Robert R. Soule, Director, Operational Evaluation Division
rsoule@ida.org • (703) 845-2482

Rigorous Analysis | Trusted Expertise | Service to the Nation

INSTITUTE FOR DEFENSE ANALYSES

IDA Document NS D-21562

Space Filling Designs for Modeling & Simulation Validation

Heather Wojton, Project Leader

Kelly Avery
Han Yi
Curtis Miller

Space-Filling Designs for Modeling & Simulation

Han Yi, Curtis Miller, Kelly Avery

June 24, 2021

Contents

Executive Summary

This document presents arguments and methods for using space-filling designs (SFDs) to plan modeling and simulation (M&S) data collection.

In modern defense testing, M&S capabilities are often critical to fully characterizing a system's capabilities. The complexity of modern military systems and the environment in which they operate means that live testing is often expensive or even impossible; certain threats or combat scenarios simply cannot be reproduced on test ranges. However, while M&S tools are undeniably valuable, to ensure they produce trustworthy results, their behavior and accuracy must be well understood in relation to their intended use.

As part of a thorough validation, testers must develop a data collection strategy for both the live environment and the M&S. This strategy includes deciding which points, and how many points, should be sampled across the combinations of possible M&S inputs, and which should be left out. In order to demonstrate the need for such a strategy, let us consider a hypothetical scenario in which the output of an M&S tool across the entire factor space is completely known to us (in reality, this "true" output will not be available). **Figure 1A** represents this true output from a hypothetical M&S tool. This choice of points is important because collecting inadequate data can result in the M&S severely misrepresenting the relationship between the factors (input or independent variables) and the response (dependent variable). This in turn can ultimately cause government sponsors to include inaccurate information in their reports and provide an incomplete picture of system performance to the warfighter.

Classic Design of Experiments (DOE) approaches, while appropriate for most live testing scenarios, are often inadequate for M&S, where most or all factors are continuous. DOE relies on relatively few samples placed on the extreme values and centroids of the parameter space and interpolated under the strong assumption of linearity of the response surface. Because this assumption of linearity is likely to fail in M&S scenarios (**Figure 1B**), testers will get better results by using SFD when planning data collection for M&S tools. SFD is a set of principled approaches to "fill" the parameter space. SFD can significantly lower the risk of mis-estimating the response surface of the model of interest by placing samples throughout the parameter space to better capture local deviations from linearity (**Figure 1C**).

Furthermore, using data collected from an SFD to build a response surface model of the M&S can provide cost savings. Typically, if testers want to obtain M&S predictions over a specific subspace, they must request the time and resources to run the M&S tool itself. If instead testers collected data across the entire factor space using an SFD, the resulting statistical model of the data could serve as a surrogate for the M&S itself, thereby eliminating the need to run the M&S over and over.

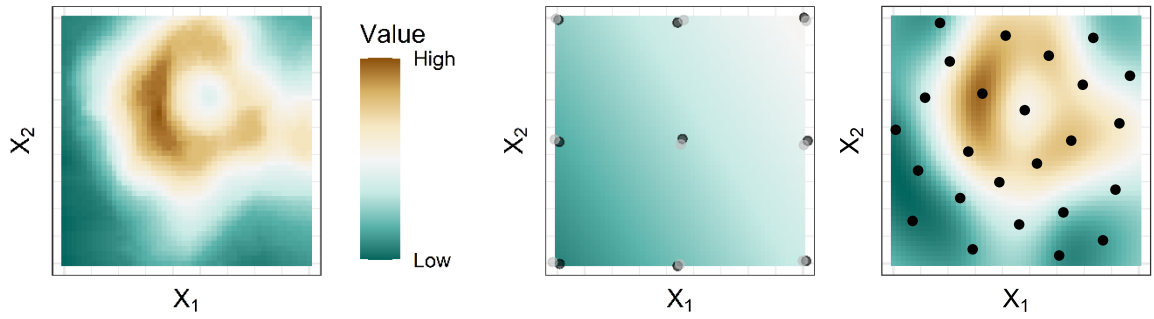


Figure 1. SFD + GP analysis (C) does a much better job of estimating ground truth (A) than classical DOE + linear model analysis (B)

M&S allows exploration of multiple variables with wide ranges of possible values. **Figure 1A** shows a hypothetical model with two input variables (X_1 and X_2). Colors represent the true model output at specific points in that 2-dimensional space. In reality, these values would not be available as ground truth but could only be partially sampled and reconstructed. In classical DOE, illustrated in **Figure 1B**, a factorial design with replication is used to sparsely place samples in the factor space, represented here as dots, followed by modeling the response surface under the assumption of linearity. Note that the modeled response surface (see color) misses the true distribution of values seen in (A), owing to local nonlinearities that violate the linearity assumption. In **Figure 1C** an SFD is used to fill the factor space with numerous samples. The response surface is modeled using a statistical emulator called the Gaussian Process (GP) model, which effectively captures major features of the ground truth values shown in (A).

The core intended audience for this introduction to SFD is those who are committed to using robust methodologies to maximize knowledge gains from M&S, but who are unfamiliar with the use of SFD. We developed a set of SFD recommendations for those who support operational testing or similar tasks, summarized in **Table 1**. We focus on providing recommendations that strike a good balance among effectiveness, readiness of implementation (i.e., "Can it be run in standard software with just a few clicks or lines of code?"), and the

amount of vetting (i.e., "Has it been around for a while, and do we fully understand its shortcomings?"). In particular, we give special consideration to the use of categorical factors (e.g., type of missile) and disallowed combinations (e.g., unrealistic angle-velocity combinations), as these are common features of M&S tools in the defense field.

Table 1. Recommended SFDs

M&S Properties	Recommended Design
All Continuous Inputs	Maximin-Latin HyperCube Sampling (LHS) or Uniform
Categorical Input(s)	Sliced LHS or Fast Flexible Filling
Disallowed Combinations	Fast Flexible Filling

The above recommendations assume M&S that is either deterministic or has a limited amount of randomness (e.g., Monte Carlo draws). There are scenarios where SFD may NOT be the best approach. For example, if an M&S has highly noisy output, a classical DOE approach may be most appropriate, otherwise we risk overemphasizing noise in our results. Alternatively, testers could consider a hybrid design approach where an SFD is combined (or overlaid) with a classical design, thus facilitating multiple types of statistical modeling.

To successfully implement a validation strategy using SFD, testers must first choose the specific type of SFD to employ based on their goals and resources, then evaluate the chosen design using objective metrics, and finally, analyze the sampled data once they are collected to estimate the overall response surface. In the sections below, we give an overview of some of the more common and useful SFDs, how to evaluate them quantitatively, and what to do once the data are collected and ready to be analyzed.

Introduction

Importance of M&S Validation

Evaluations of system operational effectiveness, suitability, survivability, and lethality increasingly rely on M&S (including but not limited to digital computer models, hardware-in-the-loop simulations, and threat emulators) to supplement live testing. In order for these supplements to be valuable, testers must understand how well the models represent the systems or processes they simulate. The verification, validation, and accreditation (VV&A) process facilitates this understanding (Wojton et al. 2019). Verification and validation are not yes/no answers; rather they involve the quantitative characterization of differences in performance metrics across a range of input conditions (Council 1998). These quantities and ranges should be relevant to the intended use of the M&S, and the resultant VV&A applies for that intended use, over those value ranges; it does not automatically extend to other uses of the M&S.

If testers wish to use results from an M&S tool in an operational or live fire test report, performing a thorough validation of the tool is critical. Validation occurs when a verified model is compared with another body of knowledge, such as live test data or subject matter expertise. Given that live data often doesn't exist across the entire operational space where testers intend to employ the model, a good validation strategy should include two broad pieces: (1) an evaluation of the model or simulation on its own across the entire intended use space, and (2) a comparison of M&S results with live test data in the regions where such data exist.

In order for testers to achieve the two validation goals above, they need a comprehensive strategy for collecting data from both the M&S and the live test. Severely deficient data from either source will seriously limit the ability of testers to draw useful conclusions from M&S output; ultimately, warfighters could be misinformed about how their equipment works. Testers typically focus on the latter validation goal of collecting a set of live test data and comparing it to a "matching" set of M&S data. For example, testers may run pre- and/or post-flight missile flyout models where they attempt to match conditions as closely as possible to the true missile flight test. This one-to-one comparison is essential for understanding how well the M&S can replicate true results in those conditions. However, the missile flyout models will almost certainly be

used in the field to predict missile performance beyond those specific conditions: in different locations, at different ranges, against more advanced threats, in coordination with other weapons, and so on. As part of a validation, testers and subject matter experts should study M&S behavior in ALL relevant parts of the operational space, even if no live test data were collected in those specific regions.

The need for SFDs

DOE techniques are commonly used in operational testing to plan live tests, where there are generally few controllable factors and the environment is usually highly stochastic. Although underlying behavior in live environments can be very complex, testers typically cannot uncover these complexities with precision because of the large amount of noise in the operational environment; a relatively simply linear or low-order model is usually the best that we can do. Classical DOE techniques (e.g., factorial, fractional factorial, or optimal designs) are designed to explain variability and estimate factor effects via the construction of statistical regression models (Montgomery 2017). These techniques tend to fill the *boundary* of the factor space because that is the best way minimize the standard errors of factor effects in a linear model. Thus, these methods are often the best strategy when including noisy live data in analysis.

However, classical DOE may not be the most effective or efficient tool for collecting data from M&S, where (1) outputs are typically much less noisy, (2) there are potentially dozens of input parameters, and (3) results may be highly non-linear. If a model is deterministic, meaning that every time the model is run with identical input values, the output is the same, then replication is unnecessary and wasteful. Note that in classical DOE, replication is a core tenet due to the noisy (non-deterministic) nature of the output. Unlike live tests where there are often few factors that can be controlled and each run may be extremely expensive, M&S inputs are typically plentiful and controllable and often it is possible to run many trials at relatively little cost. Finally, planning a design around a standard second order regression model, as is typical in live testing, may be insufficient to capture the non-linear nature of M&S output. Most classical DOE designs are boundary-filling rather than space-filling and thus are not ideal for capturing local phenomena which are common in M&S contexts.

The field of computer experiments attempts to address the unique challenges associated with collecting data from a model or simulation. Without having to worry about random variation, replication, or most practical considerations about hard-to-change factors, design points in simulation experiments are free to spread out across the design space. Computer experiments are generated to fill in the space to facilitate robust interpolations and predictions. Because of this objective to fill the space, the general class of experimental designs for computer experiments is called space-filling designs (SFDs).

For many of the M&S tools used in operational and live fire evaluations, SFDs are likely the most effective and efficient way to collect data from the model and support a complete evaluation of the model's behavior (the first of the validation goals listed above). SFDs support the creation of a statistical emulator, or meta-model, which can be used to estimate uncertainty and predict the output of the simulation in both tested and untested conditions. If the emulator is robust enough, it can also serve as a surrogate for the model or simulation itself, which can save time and cut costs by avoiding the need to repeatedly re-run the simulation.

The subsequent sections of this document provide an overview of the types of SFDs, how to evaluate such designs, how to analyze the data produced from them, and the software tools available for implementing these techniques.

Basic Space-Filling Designs

The ideal SFD for a test would cover the entire M&S factor space with an infinite number of runs. This would provide a foolproof understanding of the response surface of the model. However, given time and resource constraints, programs need a realistic way to fill the space well enough while using fewer samples. In this section, we provide guidance for choosing a better SFD for M&S with continuous factors and no constraints on the design space. Here, we caution against using General Factorial and Random designs,

which while simple and intuitive, suffer from potentially being wasteful and highly inconsistent, respectively (Garud, Karimi, and Kraft 2017; Pronzato and Müller 2012; Gramacy 2020). Instead, we recommend the use of Uniform, LHS, and maximin designs, which adhere to different design philosophies regarding what it means to achieve good coverage of the factor space, as we will further elaborate below.

General Factorial and Random

While we do not recommend their use, discussing General Factorial and Random designs enables a better understanding of what is, and is not, desirable in an SFD, and so we present an introduction to them here. One of the simplest approaches to filling a space is to place samples in a rectangular grid, guaranteeing even coverage of all dimensions for the complete range of possible values (**Figure 2A**). This General Factorial design, sometimes called Uniform Grid in SFD literature, is equivalent to multi-level factorial designs in classical DOE. The even coverage provided by this design is a desirable trait, as we do not want to bias ourselves into under- or oversampling different subspaces without justification (Damblin, Couplet, and Iooss 2013). However, because not all variables can be screened for relevance prior to M&S, this design is highly cost-inefficient (Gramacy 2020). Consider a hypothetical model with ten variables, where we decide to evenly distribute three samples along each dimension and all combinations thereof. The design would demand a total number of 3^{10} (= 59,049) samples. If only half of these variables end up predicting the response variable, then over 99% of the samples ($3^{10} - 3^5 = 58,806$; ~99.6% of all samples) would have been wasted beyond having provided the null result (see **Figure 2A** for a two-dimensional example). This phenomenon where the samples completely overlap once projected to another dimension is called "collapsing." Collapsing designs cause wastage of samples on an exponential scale to the number of discarded variables, and it is thus an undesirable trait for SFD (Janssen 2013).

To prevent collapsing, we could instead draw the samples from a random uniform distribution bounded by the parameter space (**Figure 2B**). In a Random design, it is unlikely that a large proportion of these samples will overlap once projected to other dimensions, as the locations of samples are, by definition, random across all dimensions. However, the effect of randomness works both ways: because there is no control over how the samples are placed, there is no guarantee against ending up with an undesirable design in which samples are clustered together. Thus, while the samples are drawn from a uniform distribution, our particular set of samples is likely to be non-uniform (Dalal, Stefan, and Harwayne-Gidansky 2008; Garud, Karimi, and Kraft 2017).

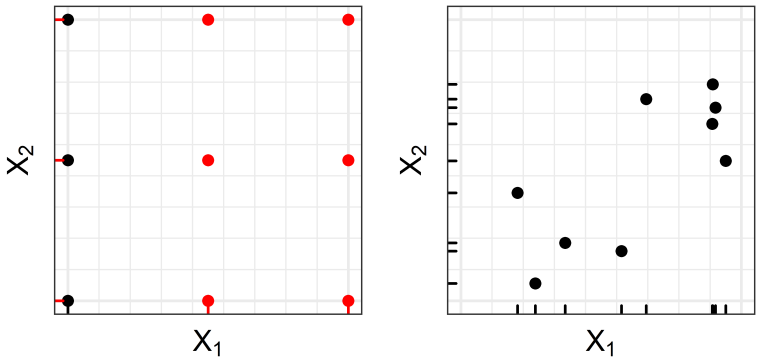


Figure 2. Examples of a General Factorial design (A) and a Random design (B)

Why simple designs should not be used.

As illustrated in **Figure 2A**, General Factorial designs provide even coverage across the space, but grid-like placement of samples is wasteful when some variables are irrelevant for the model. Here, if X_1 does not change the response variable, then samples placed along that dimension (red) become redundant. **Figure**

2B illustrates how Random designs can produce randomly placed samples that may provide poor space filling for a given iteration.

The above discussion yields a key insight into what makes a good SFD: consistency in placing samples evenly throughout a space while not collapsing across dimensions (Garud, Karimi, and Kraft 2017). There is no design that achieves this objective flawlessly, as each approach has its own unique set of trade-offs, but research in this field has produced several improved options (Pronzato and Müller 2012; Chen et al. 2006).

Latin Hypercube Sampling (LHS)

LHS is an SFD that has been in use at least since 1979 (McKay, Beckman, and Conover 1979). In its purest form, LHS places a sample in a unique position in all dimensions. **Figure 3A** visualizes an example of a two-dimensional LHS, which shows that each sample occupies a unique row and a unique column as delineated by equally-sized segments for each dimension. Even if one of the axes had no predictive power, all samples would still provide variability across the remaining axis. Thus, LHS always prevents collapsing, which is an advantage over a General Factorial design. It further guarantees uniformity for each dimension, unlike Random designs. Because of its simplicity and these strengths, LHS has been extensively used for M&S for multiple decades (McKay, Beckman, and Conover 1979; Cioppa and Lucas 2007; Damblin, Couplet, and Iooss 2013; Qian 2012; Ba, Myers, and Brenneman 2015; Husslage et al. 2011).

However, not all designs produced using LHS provide equally high space coverage. In fact, it is easy to produce a design that does not provide good space coverage while adhering to all LHS rules (Cioppa and Lucas 2007; Damblin, Couplet, and Iooss 2013). The example in **Figure 3B** is just as much an LHS as the one in **Figure 3A**, but most of the samples are distributed along the diagonal, leaving the rest of the factor space uncovered. In other words, samples with higher values of X_1, or towards the right part of the space, tend to also have higher values for X_2, or towards the upper part of the space. Because the samples' spatial locations are highly correlated across the two dimensions, any observed variations in the response variable would be just as attributable to one variable as to another, making it difficult to separate out the effects of each. Thus, the correlated design as shown in **Figure 3B** has a poor space-filling property.

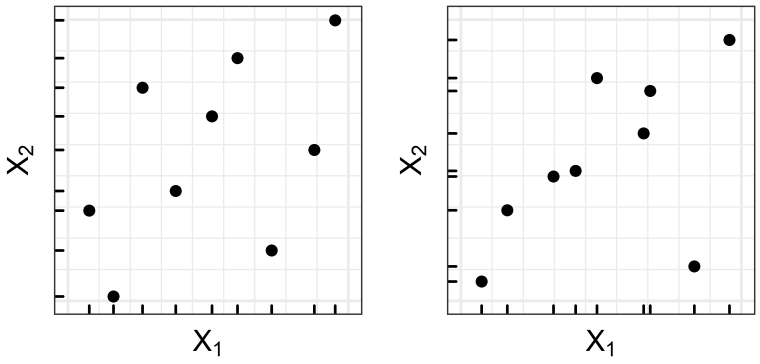


Figure 3. Two examples of LHS design

In **Figure 3A**, the samples occupy unique rows and columns. Because of this constraint, LHS designs are always uniform in each dimension. In the undesirable design example of **Figure 3B**, the samples mainly occupy the diagonal and leave the rest of the space empty. Despite the high correlation of samples' locations along the X_1 and X_2 dimensions, this design satisfies the LHS criteria perfectly.

Maximin

General Factorial and LHS designs use explicit rules ("place samples along a rectangular grid"; "place samples in unique row-column cells"), which, as long as they are satisfied, provide no further information about the properties of the design. In contrast, maximin design seeks to find the best placement of samples in the

parameter space that maximizes a quantifiable criterion: the minimum pairwise distance across samples (Johnson, Moore, and Ylvisaker 1990). Geometrically, maximin as SFD is equivalent to finding the largest equal-sized spheres that can fit within the design space, where their diameter corresponds to the criterion value (Pronzato and Müller 2012; **Figure 4A**). Simply put, a maximin design guarantees that all samples are spaced apart from one another by at least the value of the maximin criterion (Johnson, Moore, and Ylvisaker 1990).

The maximin criterion is not only used to generate a design, but also serves to directly quantify its goodness: for a given sample size, the larger the value, the better the design is from the perspective of maximin. Thus, it is possible to generate a number of maximin designs and compare their goodness (Garud, Karimi, and Kraft 2017). The maximin criterion, however, only provides the floor for the distance between the closest pair of samples, providing an incomplete picture of their space-filling properties (Pronzato and Müller 2012; **Figure 4B**). Moreover, the distance-based criterion systematically biases the design to have more samples near the boundary rather than the center of the parameter space, resulting in non-uniform coverage (Gramacy 2020).

Despite these flaws, maximin is a very useful tool for both evaluating and optimizing a design. As such, it can be useful in further optimizing a design that was initially created using other approaches, such as LHS (**Figure 4C**; Damblin, Couplet, and Iooss 2013).

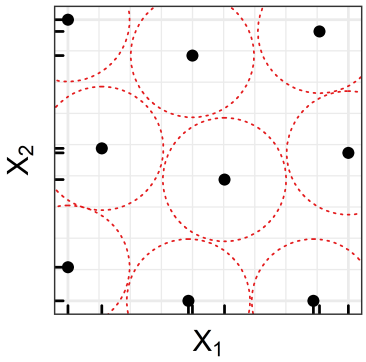


Figure 4A. Maximin design can be visualized using spheres

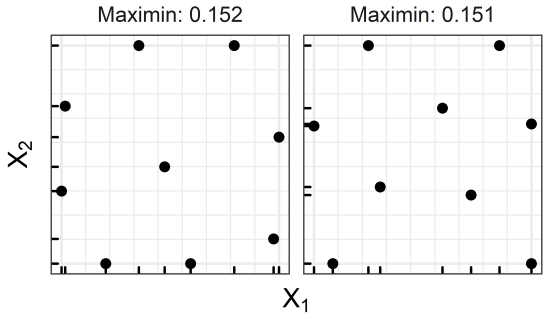


Figure 4B. Two Maximin designs can be compared for 'goodness'

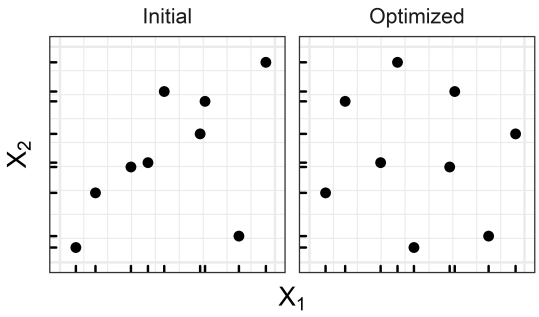


Figure 4C. LHS design (left) can be optimized (right) using maximin

Figure 4A visualizes the concept that maximizing the minimum pairwise distance is equivalent to packing spheres (red circles) in the design space. **Figure 4B** shows two maximin designs with similar criterion values. Comparing the two designs shows that optimizing the worst-case scenario does not guarantee even spacing. The two plots in **Figure 4C** illustrate how maximin can be used to optimize existing designs. In this case (the initial LHS design (left) is shown after being optimized with maximin (right).

Uniform Designs

Uniform designs seek to scatter points uniformly across the domain, without relying on a simple grid layout (see **Figure 5**). First developed in 1980 (Fang 1980), Uniform designs require a one-dimensional balance across all factors, just like LHS. However, they also impose an additional constraint by requiring multi-dimensional uniformity (Fang et al. 2000). Intuitively, uniformity is achieved when the density of samples is constant across the factor space. Thus, uniform design prevents undesirable instances of LHS designs where large portions of the space are left empty (e.g., **Figure 3B**). Formally, uniformity is achieved by optimizing a metric called L_2-discrepancy, which is discussed later in Section 5 and the Appendix. In this section, we note that for future reference the uniform design can be viewed as a discrepancy-optimized LHS.

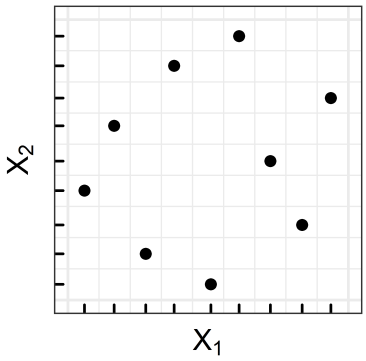


Figure 5. Uniform Design

In **Figure 5**, which shows an example of Uniform design, samples are uniform in single dimensions, as they occupy the center of each cell for each unique row and column, as in a typical LHS. However, the design is also uniform across both dimensions (i.e., "density" of samples is even throughout the factor space), which is not a necessary trait of LHS.

Special-Purpose Designs

In defense testing, M&S tools frequently have characteristics that are either incompatible with the conventional designs described above, or make those conventional methods less ideal; these problematic characteristics are:

- Categorical factors
- Disallowed combinations
- High computational or resource cost

Categorical factors

Operational and live fire tests often involve variables that are categorical (such as missile type) rather than continuous (such as missile velocity). Because it is not possible to define numerical distances across the levels of a categorical variable like 'vessel type', a conventional SFD cannot be applied out of the box[1] (Yulei Zhang and Notz 2015). One solution is to create a separate space (or "slice") for each level of the categorical factor, and define each slice by all the *continuous* variables (this also works with multiple categorical variables: simply create a slice for each unique combination of levels) (Qian 2012; Ba, Myers, and Brenneman 2015; Lekivetz and Jones 2019). SFD can then be applied to each slice. To ensure that values can be compared across slices, and that samples will not be wasted if one or more levels are found to be not predictive, it is desirable that the design retains similar space-filling properties both within and across the levels. Sliced LHS and Fast Flexible Filling designs (introduced in the following sections) are two options for handling categorical factors in an SFD.

Disallowed combinations

Another common characteristic of defense testing is that some parameter values, or combinations of values, have little operational value or are physically nonsensical. Simulating the effects of supersonic flight on the F-35 jet at 1,000 ft above the sea level, for instance, may not be a valuable use of effort. In this case, it could be prudent to "block off" a region of the parameter space accordingly, but doing so can create a non-rectangular parameter space. In practice, this space blocking can be done ad hoc by first creating an SFD and then rejecting all samples falling within the rejected region. The problem is that this approach will invalidate the optimality achieved by SFD as assessed during its creation[2] (Lekivetz and Jones 2019). Therefore, in cases where it is crucial to optimize and maintain a design's space-filling properties for a model with a significant proportion of disallowed combinations, we need to use SFD that can natively deal with non-rectangular spaces (Draguljić, Santner, and Dean 2012). Fast Flexible Filling and Minimum Energy designs, discussed in the following sections, are two options for dealing with disallowed combinations or design constraints.

High computational or resource cost

The SFDs discussed so far share a core trait: they are all agnostic to the particular study to which they are applied. This means that if one developed a good LHS design for a ship damage model with 12 dimensions and 50 samples, the same design could then be directly applied to a cybersecurity model with 12 dimensions and 50 samples. Sometimes, however, testers may wish to develop a design that is tailored to the dataset in

[1] It is important to NOT assign arbitrary numbers to each level (e.g., 0 for submarines, 1 for cruisers, 2 for destroyers) to "trick" SFD into thinking that categorical variables are in fact continuous. If so, SFD will assume absurd relationships across the levels (e.g., cruisers is one more than submarines, and MANSUP is twice destroyers) and create an "optimal" design accordingly.

[2] Another ad hoc solution is to reject all designs that have any samples within the disallowed region. The optimality evaluation will still be invalid, however, unless the evaluation method properly takes the non-rectangularity into account.

question. In particular, M&S with high computational cost may benefit from focusing a subset of samples strategically within a narrower subset of the parameter space where data may behave in an interesting manner (e.g., large prediction errors indicating local nonlinearity). In this case, testers may benefit from using Adaptive Sequential designs, discussed in the following sections.

Sliced LHS

Regular LHS creates a design in which all samples occupy unique rows and columns in a space defined only by continuous variables. Sliced LHS further extends this principle to individual slices (Qian 2012), by creating what is akin to "nested" LHS. For example, consider a hypothetical scenario where we wish to study the effects of the aesthetics of a button (e.g., green circle, orange triangle, and purple square) in addition to those of continuous variables. We first create an LHS for each aesthetic-slice with the desired number of samples (e.g., $N = 4$; **Figure 6A**). When these slices are overlaid, each of the four rows and columns will contain three samples (**Figure 6B**). To turn this overall design into an LHS, we need to further subdivide each row and column into three segments (corresponding to the number of slices) and assign every sample to a unique segment (**Figure 6C**). Since Sliced LHS is a variant of LHS, it inherits the same core strength (one-dimensional uniformity) and weakness (no guarantee against correlated designs), although optimization process can help with the latter (Ba, Myers, and Brenneman 2015).

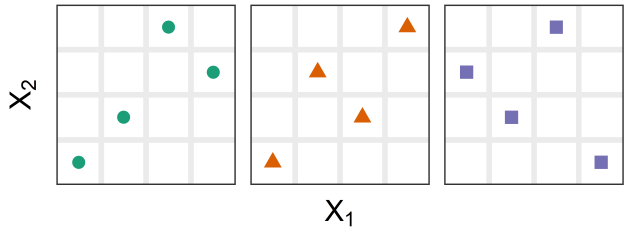


Figure 6A. Creation of Sliced LHS (SLHS) begins with creation of unique LHS for each categorical slice.

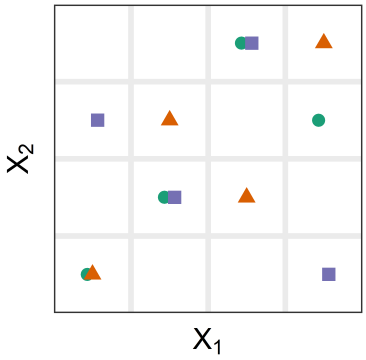


Figure 6B. Without further adjustments, the overlaid design is not LHS in the overall parameter space defined by continuous variables.

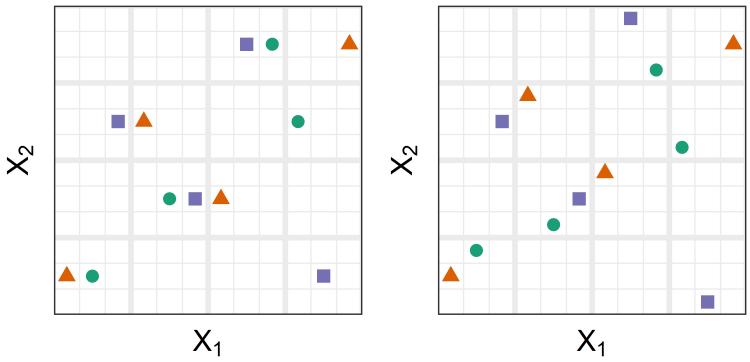


Figure 6C. Sliced LHS is generated by assigning samples within the primary grid (thick lines) to unique positions along (Left) each column and (Right) each row (thin lines).

A sliced LHS design preserves one-dimensional uniformity in the overall space defined only by continuous variables, as seen in **Figure 6A**, as well as within individual slices representing unique levels of categorical variables, as illustrated in **Figure 6B**.

Fast Flexible Filling (FFF)

The overarching goal of FFF is to generate samples that are the closest to all points of the factor space.

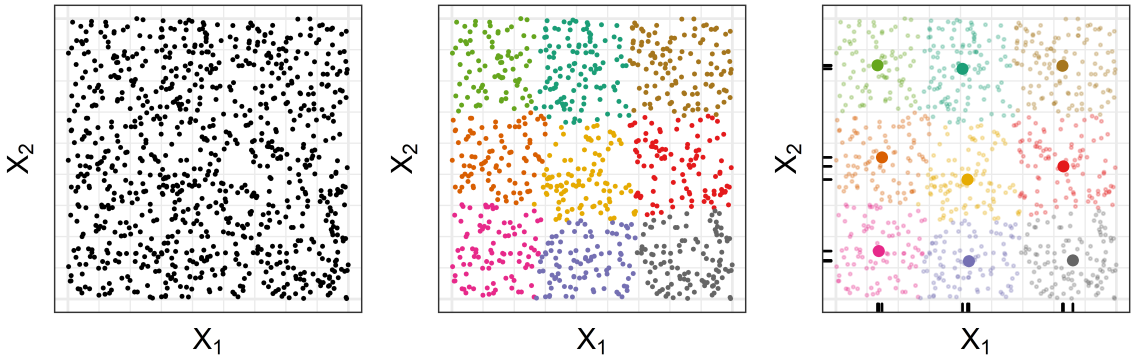


Figure 7. Schematic description of the clustering method used in creating FFF designs.

The first step, shown in **Figure 7A**, is to randomly generate many points, which serve as a representative subset of all possible locations within the factor space. Next, the random points are grouped into spatial clusters, as shown in **Figure 7B**, where the number of clusters equals the number of desired samples. Finally, a sample is placed within each cluster, usually at the centroid, as shown in **Figure 7C**.[3] Thus, any given point within the parameter space is reasonably close to a cluster, and any given point within the cluster is reasonably close to a sample (Lekivetz and Jones 2015).

For designs with categorical variables, FFF adds one more step where secondary clusters are formed within each primary cluster of the initial random points. Then, these secondary clusters are assigned to different slices (Lekivetz and Jones 2019). The resulting samples should still be representative both of the overall space, as shown in **Figure 8A**, and of each categorical slice, as shown in **Figure 8B**.

[3]These steps are for illustrative purposes only, and the actual values used and the clustering scheme do not represent the actual FFF as implemented in JMP.

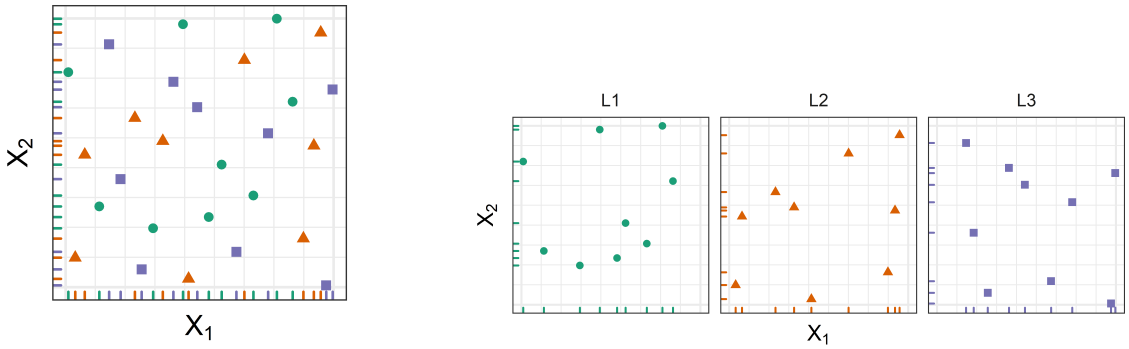


Figure 8. FFF design retains optimality both in the overall parameter space (**A**) and within each categorical slice (**B**).

One of the most important strengths of FFF over Sliced LHS is that it can also incorporate disallowed regions during the creation of a design. As discussed earlier, defense testing often involves combinations of variables that are not informative (e.g., low altitude and high speed for a jet). Sliced LHS does not natively provide a way of distributing samples across categorical slices while taking the disallowed regions into account during its creation. FFF, on the other hand, can place representative samples across multiple non-rectangular regions (Zhu et al. 1996; Zemroch 1986), even when some variables are categorical. **Figure 9** shows how no samples (black dots) are placed in the disallowed (dark) region when creating the FFF design.

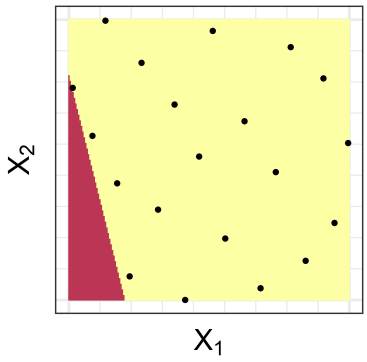


Figure 9. FFF can incorporate disallowed combinations

Minimum Energy

Minimum Energy is a highly flexible design that can accommodate irregular test regions (Joseph, Gul, and Ba 2015; Joseph et al. 2019). It is especially useful when the distribution of test values across the parameter space is continuous rather than binary. In Minimum Energy, samples are treated as electric charges of equal sign that repel one another. The goal is to reach an equilibrium where the potential energy is minimized by moving the samples around (Joseph, Gul, and Ba 2015). The resulting placement of samples is radial.

Importantly, locations across the factor space can be assigned with different magnitudes of electric charge (Joseph et al. 2019). Minimum Energy design places samples based on the desired density function across the parameter space, effectively creating a continuum of test values of regions. Consider the model in **Figure 10** where the tester has mapped out how informative an M&S run would be across the space based on prior knowledge. For example, a mortar round may rarely be fired at a really close range, and thus M&S resources need not be spent on extensively sampling that region of the factor space, which has relatively lower test value. At the same time, however, it may still be preferable to have some information about the rounds' behavior rather than none. Minimum Energy design could be a good choice in this case, where instead of

strictly prohibiting samples from occupying disallowed regions, samples are concentrated in areas with higher test values and sparsely placed in areas with lower test values.

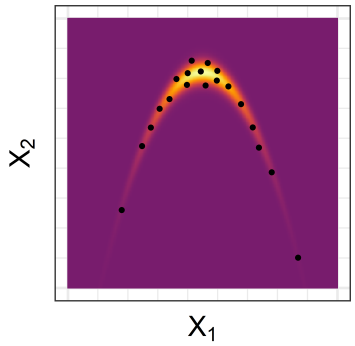


Figure 10. In Minimum Energy design, final placement of samples (black dots) is concentrated in regions of higher test values (bright) and avoids regions with lower test values (dark)

Design Evaluation

Just as with any design strategy, we need ways to measure if a certain SFD is "good." In classical DOE, metrics such as power and confidence (among others) are commonly used to assess goodness. However, in the M&S paradigm where replication can be irrelevant and the goal of the design is to efficiently fill the design space, we need different criteria for determining whether an SFD has good properties. While there are many criteria to consider, it is particularly important that an SFD satisfy the following three criteria in order to be useful:

- **Point-distance:** Samples are placed as far apart from each other as possible.
- **Uniformity:** All regions of the design space are equally well represented.
- **Projection:** The design is robust to variables being collapsed.

While these properties may be informally assessed in simpler cases, visual inspection of a design becomes increasingly challenging as the number of variables grows beyond three. Furthermore, an objective metric can help analysts compare classes of designs, compare sample size options within a specific design type, and ultimately justify a particular design choice to an external party such as a sponsor.

Space-filling criteria

Multiple metrics exist for assessing the three criteria above. Because these metrics differ in how they formally define the property, a property (e.g., uniformity; point-distance) assessed using one metric (e.g., discrepancy; maximin) may differ significantly when assessed using another metric (e.g., minimum spanning tree; minimax). In this document we recommend one metric for assessing each property; however, it never hurts to look at multiple metrics to get a more holistic feel for a design's performance.

Below we summarize the three primary evaluation criteria mentioned above, and provide specific recommended metrics that testers can calculate to assess a particular criterion and compare designs (see **Table 2**). Further detail, along with additional criteria and metrics, are provided in the **Appendix**.

Table 2. Recommended metrics for evaluating space-filling properties.

Property	Metric
Point-distance	Maximin
Uniformity	Centered L^2-discrepancy
Projection	MaxPro

Point-Distance

When creating an SFD, we want to create a design where the samples are not too close to one another (which may introduce redundancy), but also not too far away from one another, which may result in undersampling of a subspace. Point-distance metrics calculate the distance across samples or points, providing partial insight into the extent to which the placement of samples is optimal.

Maximin (also referred to as "mindist") is one way to measure point-distance. This metric calculates the minimum pairwise distance between samples of the design (see the **Appendix** for the equation), where higher values indicate a better design. Maximization of this criterion can be used as an optimization criterion for a design as seen in the maximin-optimized LHS design above.

As with any other evaluation criterion, maximin should not be used by itself to evaluate a design, as it provides a limited picture. maximin only quantifies the worst-case scenario: the minimum pairwise distance; it supplies no information about how far apart the other pairs of samples are spaced. Also, designs with samples placed close to the boundary can have higher maximin values, causing undersampling towards the center of the parameter space. That said, maximin can be a useful way to evaluate a design, as long as one keeps in mind its limitations.

Of the basic designs presented above, the General Factorial, Uniform, maximin, and LHS (optimized with maximin) designs all perform relatively well in regards to the point-distance criteria. LHS and Random designs generally do not satisfy this property.

Uniformity

Uniformity refers to the property of equal representation of all regions in the parameter space. One way to evaluate uniformity is to assess the "discrepancy" between the proportions of numbers of samples included in the subspaces of a design and those of their volumes. Design points which sufficiently approximate a uniform distribution will exhibit low discrepancy between the two proportions across subspaces, and vice versa.

Consider the two examples in **Figure 11**. The two space-filling design examples (LHS and LHS-maximin) have identical numbers of dimensions (X_1 and X_2) and samples (black dots). However, this does not indicate that the two designs are similarly uniform. To highlight this insight, we may focus on two arbitrary subspaces of a size within each design space (see red- and blue-shaded areas). In the Random LHS design, the percentage of samples in each example subspace deviates from the percentage of area covered by that subspace. This discrepancy is lower in LHS-Maximin, which suggests that the latter is more uniform. These calculations can be performed using multiple subspaces to assess the overall extent of discrepancy between the expected density of samples, which can then serve as a metric for uniformity: higher discrepancy (density of samples is unequal across the factor space) indicates lower uniformity, and lower discrepancy (density of samples is equal across the factor space) indicates higher uniformity.

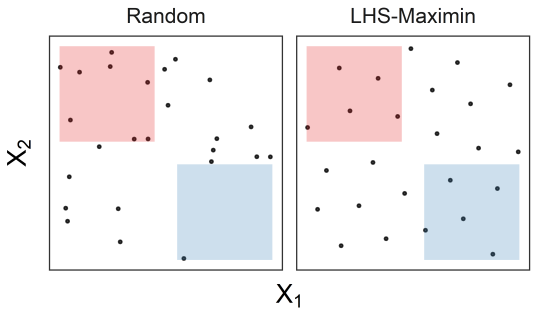


Figure 11. Less discrepancy across factor space in LHS-Maximin example indicates better uniformity than in Random LHS design

There are many variants of discrepancy metrics depending on how the summary value is calculated. Star-discrepancy calculates the maximum magnitude of the difference in proportion of samples and proportion of volumes (Damblin, Couplet, and Iooss 2013). In practice, however, calculating the maximum of all possible subspaces is computationally intensive. L^2-discrepancy—which is easier to compute as it uses the L^2 norm—serves as a surrogate for the more ideal metric; we thus use it instead to evaluate uniformity (or a variant of it, such as centered L^2-discrepancy). We have relegated a deeper, mathematical discussion of the metric to the **Appendix**.

As the name implies, Uniform designs have excellent uniformity properties, as do LHS-Maximin. Conversely, LHS and Random designs are likely to perform poorly in a uniformity assessment.

Projection

Projection is a property which describes how samples are placed to provide information along as many variables as possible in a given design. This property is important especially when M&S involves a large number of variables, since not all of them may turn out to affect the response. A design with good projection ensures that samples of a space-filling design continue to provide robust coverage of the remaining variables even if a subset were to be discarded during analysis. In contrast, samples in a design with poor projection may "collapse" onto a single value or similar values when projected to the remaining variables, in which case we would have effectively wasted samples. While an ideal metric would assess projection properties for all possible subspaces defined by any combination of variables, such a metric would be extremely time-consuming to compute.

The maximum projection (MaxPro) metric bypasses this computational difficulty (Joseph, Gul, and Ba 2015). Intuitively, an SFD with good projection property will satisfy the following conditions: (1) no two samples will ever completely overlap in any individual dimension, and (2) all dimensions will achieve a reasonable degree of uniformity. In **Figure 12A**, the two samples are distinguishable along one dimension (X_1), but completely overlap in the other (X_2), in a phenomenon referred to as "collapsing." The set of samples in **Figure 12B** are uniformly spaced when projected to one dimension (X_1), but are relatively more clumped when projected to the other (X_2).

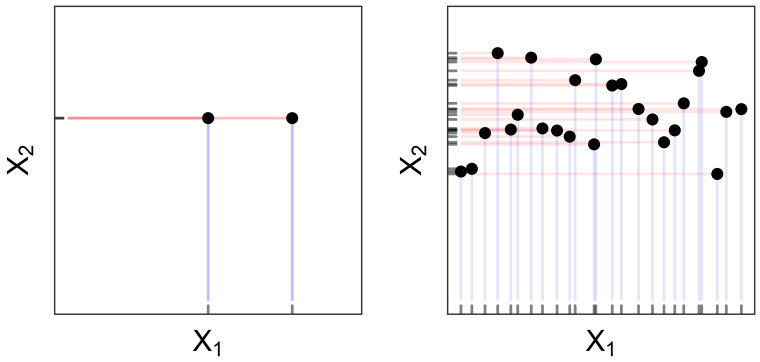


Figure 12. High MaxPro values indicate undesirable projection properties

Lower MaxPro values imply better projection properties. This is because MaxPro is inversely scaled by the product of the square of all pairwise distances in each dimension. A design with poor projection properties that fails to meet either of the two conditions (non-overlap and uniformity) will produce higher MaxPro values. For instance, any collapsing pair of samples will turn the distance product to 0, causing MaxPro to reach infinity. In a less extreme example, lower pairwise distances in any of the dimensions will result in lower values for the distance product, driving the final MaxPro value to be higher (see the **Appendix** for more details).

LHS and Uniform designs are explicitly constructed to perform well in terms of projection. General Factorial and Random designs, on the other hand, have poor projection properties. Maximin designs neither excel nor perform poorly with respect to this metric.

Using the evaluation criteria

While we have thus far focused on the comparison across design types (e.g., LHS vs. maximin), evaluation criteria are also important when comparing across multiple iterations of the same design type. Many SFDs are created stochastically, which means that each time a design is created, it may exhibit considerably different properties.

In **Figure 13**, we show how six different design types—General Factorial, Random, LHS, maximin, LHS-Maximin, and Uniform—score on two metrics: maximin (point-distance) and centered L^2 discrepancy (uniformity). The two-dimensional space of the plot is defined by maximin on the x-axis (further right is better) and centered L^2 discrepancy on the y-axis (lower is better). Multiple iterations were created for the four stochastic designs (100 each for Random, LHS, LHS-Maximin, and maximin), to better illustrate the variability from one iteration to the next, and we plotted one iteration each for the General Factorial and Uniform designs. Among stochastically created designs, the Random design exhibits the greatest amount of variability across iterations, while LHS-Maximin shows the least. All designs involved two dimensions and nine samples. As the figure shows, testers can use these measurements not only to choose the type of design but also to select the better performing iterations within the desired design type.

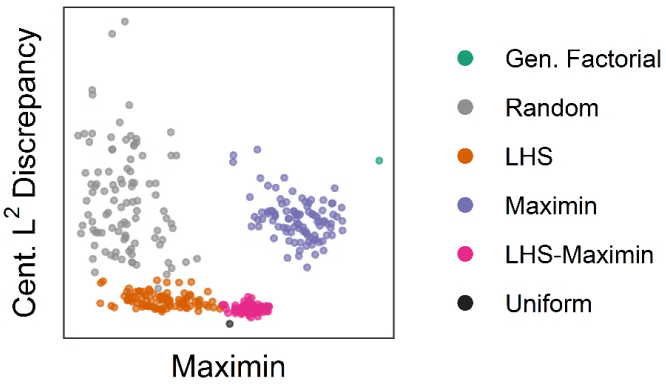


Figure 13. Point-Distance and Uniformity Comparisons Among Designs

Additionally, after the data are collected, model-based criteria can be used to assess the goodness of fit and the degree of uncertainty of a probabilistic model (discussed in the next section). These metrics are similar to optimality criteria in the classical DOE world, but more difficult to compute and optimize (see the **Appendix** for more information).

Determining the sample size

Sample size determination is a key aspect of planning any test. In M&S, if we start with an initial size that is too small, we may not capture the true behavior of the response surface, and these inaccurate predictions can lead us to focus on inappropriate regions of the parameter space or report misleading results. Conversely, too large of a sample size is wasteful, especially when the computational budget is limited (Liu, Ong, and Cai 2018). While important, the literature currently lacks clear formal guidance for choosing the right sample size for deterministic computer experiments, unlike for classical DOE. In the absence of a simple one-size-fits-all solution, we offer several ideas for how to approach sample size determination for SFDs.

As a simple rule of thumb, the number of runs (N) for an effective initial computer experiment should be at least 10 times the number of variables (d), i.e., N=10×d (Loeppky, Sacks, and Welch 2009). If resources are especially limited, such that this minimum cannot be attained, testers could instead plan for 10 times the expected number of *active* variables, as the sparsity of effects principle suggests that systems and processes are often dominated by relatively few main effects and low-order interactions. Note, however, that this rule provides the lower bound, rather than the upper bound; sample sizes based on the N=10×d rule are especially likely to prove insufficient when probing more complex response surfaces. This rule of thumb is also intended for situations where most or all factors are continuous. Cases with several categorical variables would require more samples. In that situation, multiply the number of samples used for a single category by the number of categories being investigated to get a minimum for N.

A more rigorous approach involves examining how the values for evaluation criteria change as a function of different sample sizes for a given design. The goal would be to find the optimal point of diminishing returns (the "knee" or bend point in the curve) and choose that as the sample size. The example plot in **Figure 14** shows how the evaluation criterion changes with the increasing sample size. In this case, several candidate LHS designs were created with two input variables and with sample sizes ranging from 4 to 40. Then, we calculated the L^2-discrepancy metric for each sample size. In this example, increasing the number of samples changes the metric little once we reach 15 or 20 design points. In the absence of any other criteria to judge what the sample size should be, looking for the point at which the metric's rate of improvement slows significantly may provide guidance.

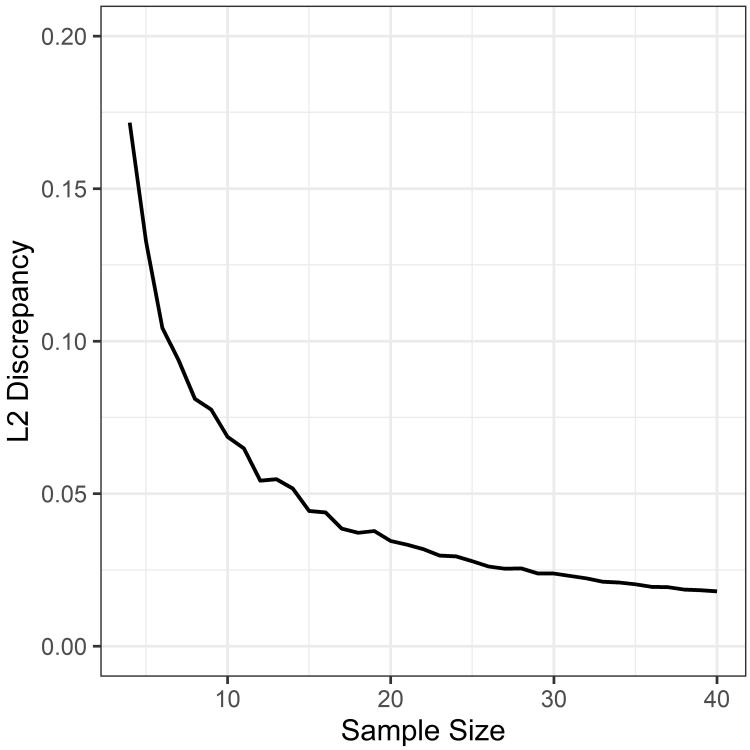


Figure 14. Scree plot of L^2-discrepancy vs. sample size for a 2-factor LHS design

We acknowledge that in defense testing, there are often practical considerations, such as the time or cost required to run a model, that may prohibit sample sizes above a certain number. The extent of these limitations should be articulated as part of an M&S validation effort. In some cases, it may be preferable to reduce the dimensionality of the experiment by not varying selected inputs in order to meet the above sample size recommendations, rather than to keep all inputs in the experiment with a deficient sample size.

Finally, subject matter experts' expectations for model behavior in response to changing inputs may further inform sample size selection. Relatively smaller sample sizes (such as N=10×d) can be justified when largely linear and low-order effects are expected. Conversely, if highly non-linear response surfaces are expected, increasing the number of samples to fill the parameter space is even more crucial.

Analysis of M&S Data

In this section, we discuss how one can analyze M&S data collected from an SFD using substantially different approaches from those typically used for analyzing live data. In live testing, outcomes are subject to stochastic variations which may arise from unseen or uncontrollable variables. This constraint often necessitates conservative analytic approaches such as linear regression to avoid overfitting to random noise at the expense of the ability to estimate a complex response surface (**Figure 1B**). In contrast, outcomes in M&S are generally deterministic, with all effects theoretically visible to modelers. If each data point can be assumed to be a true outcome, then estimating the response surface becomes a question of appropriately interpolating values for the unobserved points in between sampled data points. Such models can be highly flexible and are thus capable of estimating non-linear response surfaces, as we demonstrated earlier (**Figure 1C**).

In this section we briefly outline some methods for building non-linear models for M&S data. We first discuss classical interpolation techniques, such as splines and higher-order polynomials, which can be highly useful for modeling the M&S space. However, these techniques fundamentally suffer from the inability to quantify

the uncertainty in the unobserved regions. We next introduce GP regression, which provides a way to both model expected behavior using interpolation and estimate uncertainty in unsampled regions. In the context of M&S validation, understanding this uncertainty may be critical for assessing how well the M&S emulates the real world. For this reason, we recommend GP modeling for analyzing M&S data in most situations.

Interpolation

The objective of interpolation is to predict the values for a function in unobserved points between sampled points. Consider a collection of points along an experimental variable where response function values were observed during testing (**Figure 15**).

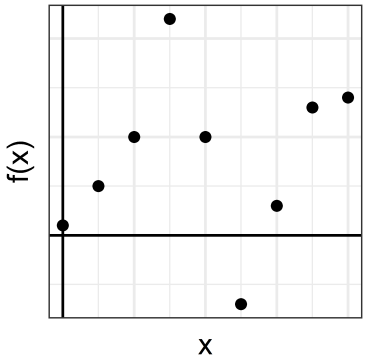


Figure 15. A hypothetical response function observed at evenly spaced points

For ease of illustration, examples in this section assume one, rather than two, dimensions for the M&S space.

If testing is not deterministic (e.g., live data), a standard way to create a function that returns values for the unobserved points is to conduct linear regression (**Figure 16A**). In deterministic M&S, there is no randomness to produce noise, so the outcome will always be the same as long as the input variables remain the same. In this case, the observed values can be leveraged as known anchors from which missing values can be interpolated. Stochastic M&S may be treated like live data if random noise tends to dominate outputs, or it may be treated as being in between live data and completely deterministic systems if the results are random but the random noise is small relative to general trends.

Intuitively, we could simply connect the observed values (when results are deterministic) with straight lines using a linear interpolator (**Figure 16B**). While a linear interpolator may do a reasonable job of predicting the unobserved values, it may not accurately resemble the true function, especially if that function is smooth. One alternative that returns a smooth function is to use cubic splines, which are piecewise polynomials that are fit in between successive observed values (**Figure 16C**). It is also possible to fit a single higher-order polynomial function to the data, but we do not recommend this method for two reasons. First, even if the test itself is deterministic, the true response function may have local variations that can exert undue influence on the overall fit of the single function. Second, polynomials are susceptible to wild oscillations if the sample points are evenly spaced ("Runge's phenomenon").

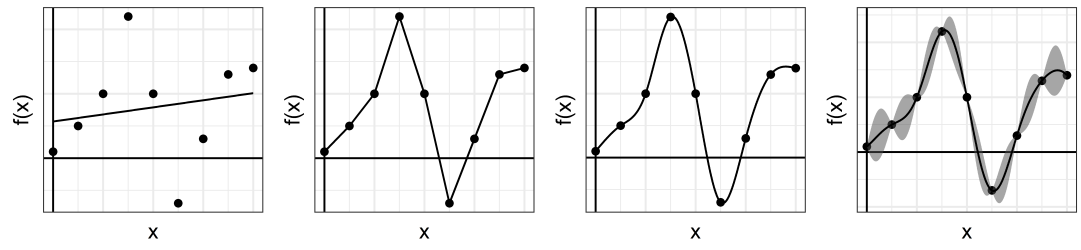


Figure 16. (A) linear regression fit; (B) linear interpolator; (C) spline interpolator; (D) Gaussian Process model, all fit to a hypothetical function.

This discussion of the different interpolation techniques reveals a key insight: even though unobserved data can be estimated, they remain unknown unless further tests are run. Thus, we can never remove uncertainty in those regions with analytical techniques. Classical interpolation techniques cannot quantify this uncertainty; rather, the interpolated points are given the same credence as the observed points. In the next section, we discuss GP regression, which remedies this fundamental shortcoming (**Figure 16D**).

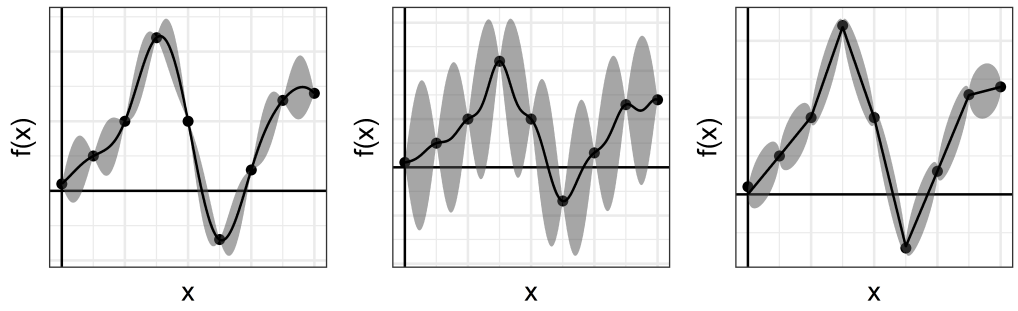
Gaussian Process (GP) Regression

Here, we discuss GP regression, which is used not only to interpolate unobserved values, but also to provide a framework to quantify the uncertainty associated with each value. Uncertainty quantification (UQ) is important for M&S validation so that testers can more accurately assess the "goodness" of M&S. For example, we may want to see if a live data point falls within the uncertainty bounds of the M&S output, or compare the uncertainty bounds themselves for live and M&S data.

UQ can be achieved by assigning a probability of matching the true response function to multiple interpolation functions. For M&S systems, a popular means of providing this UQ is through conducting GP regression, a process known as kriging. Under the right conditions, kriging constructs an interpolator that, according to the probability model, is the best linear unbiased predictor (BLUP; see (Santner, Williams, and Notz 2018)) for the value of the response function at unobserved points. GPs serve as powerful interpolators. If the assumptions of the model (e.g., covariance function) are correct, the interpolator has the smallest squared prediction error of any linear interpolator using the observed points. Importantly, the GP model allows us to endow our predictions with intervals describing the uncertainty of the value of the function at unobserved points

Because GPs come with an enhanced framework, it is important to be aware of its mechanics and caveats. GPs are probability models that describe randomly generated functions which follow a Normal (i.e., "Gaussian") distribution. As such, the properties of the uncertainty estimates from GP are determined by two parameters: the mean function and the covariance function. In practice, the mean function is rarely interesting; practitioners generally set it to the zero function. Instead, practitioners generally focus on the covariance function, which quantifies the variance of the response surface at a single point, as well as the correlation of the values of the response function between two points.

As **Figure 17** illustrates, the shape and magnitude of the resulting interpolator and uncertainty estimates depends greatly on the choice of covariance function.[4] The grey semitransparent regions represent 95% credible bands for the value of the underlying function at unobserved points. Examples A and B show kriging results when the covariance functions differ only by one parameter's value, while C shows the results when a function of a completely different family of functions is used.[5]



[4]Different families of covariance function exist and differ based on parameter values. The Matern family of covariance functions seems popular due to its flexibility, though the authors of this document are not familiar enough with this issue to offer a more emphatic recommendation.

[5]For more information on kriging, see (Wojton et al. 2019), (Gramacy 2020), or (Santner, Williams, and Notz 2018).

Figure 17. Kriging the same samples, but with different covariance functions.

GP regression is widely accepted as the "standard" tool for analyzing data from SFD and for building meta-models. The suite of techniques is flexible and allows users to quantify uncertainty across the space. Thus, we recommend using this technique for analyzing M&S data, *unless*:

1. **More than ~2000 data points are collected from the M&S.** Running a GP regression becomes computationally prohibitive with larger sample sizes (Gramacy 2020). In this case, using classical interpolators may be the best option. Alternatively, where UQ is particularly important, testers can perform GP regression on subsets of the design space.
2. **Trends are linear.** If relationships between input and output variables in an M&S are linear or only contain very low-order curvature, then there is no reason to fit a non-linear model, let alone use GP regression. In this situation we recommend using traditional linear regression techniques.

Finally, designs with categorical variables can be analyzed with latent variable GP models (LVGP; Yichi Zhang et al. 2020). This approach is based on the assumption that the effect of categorical variables (e.g., missile type) on the outcome response physically manifests via a number of underlying continuous variables (e.g., nose cone geometry, pliability of the material). If these continuous variables are accessible, they can be entered into conventional GP analysis in lieu of adopting a specialized approach for categorical variables.

Binomial Responses

The above sections focused on techniques appropriate for continuous response variables where the underlying response function is itself assumed to be continuous. Binomial response variables, such as hit/miss, are not continuous variables nor can they be connected by a continuous function.

If the M&S system is completely deterministic (setting inputs at the same values will always produce a hit or always produce a miss, for example), a simple and reasonable interpolator is the nearest neighbor interpolator, where the predicted outcome at an unobserved point in the factor space is the outcome at the nearest observed point. Often, though, the system is not deterministic and the variable of interest is actually the probability of a hit given a combination of factors. In such a situation, we are no longer performing interpolation *per se*.

A conventional approach to analyzing a random binomial response is logistic regression, which gives the probability of an outcome based on a linear model. This approach may still be appropriate and a reasonable default, but practitioners may believe that M&S systems explore such a large space of inputs that a linear model is no longer appropriate. Where there is local variation throughout the factor space not well described by a linear model, including quadratic and interaction terms in the linear model can allow for some more flexibility, and if that is still not enough, more advanced methods based on generalized additive model (GAM) or GP techniques can be employed.[6]

Summary and Recommendations

Summary

Space Filling Designs and associated analysis techniques are useful tools for M&S validation. Computer model output is often less noisy and has more controllable inputs than a live test in the physical world. Because we can capture M&S output in higher quantities without the noisiness of a live operational environment, the results of M&S data collection often appear more complex and detailed than the results of live tests. Thus, classical DOE is not always the best approach for collecting data from such a model. Instead, SFDs aim to fill the input space in order to facilitate precise interpolation across the output surface.

[6]For more information, see (Wood 2017) or (Choudhuri, Ghosal, and Roy 2007).

Many classes of SFDs are available for use in cases where there are no categorical factors or constraints on the design space. These basic SFDs include General Factorial and Random designs, Latin Hypercube Sampling Designs, maximin designs, maximin-optimized LHS designs, and Uniform designs.

M&S situations that include categorical inputs, have design region constraints, or are computationally burdensome require special-purpose SFDs. Fast Flexible Filling, Sliced LHS, Minimum Energy, and Adaptive Sequential designs represent good options for one or more of these special cases.

To assess the goodness of an SFD, we recommend focusing on three broad properties: Point-Distance, Uniformity, and Projection. As summarized in **Figure 18**, each class of basic SFD performs relatively better or worse across each category of evaluation criteria, according to our recommended metrics. Note that this plot depicts a qualitative assessment for the purpose of quick visual comparisons across techniques.

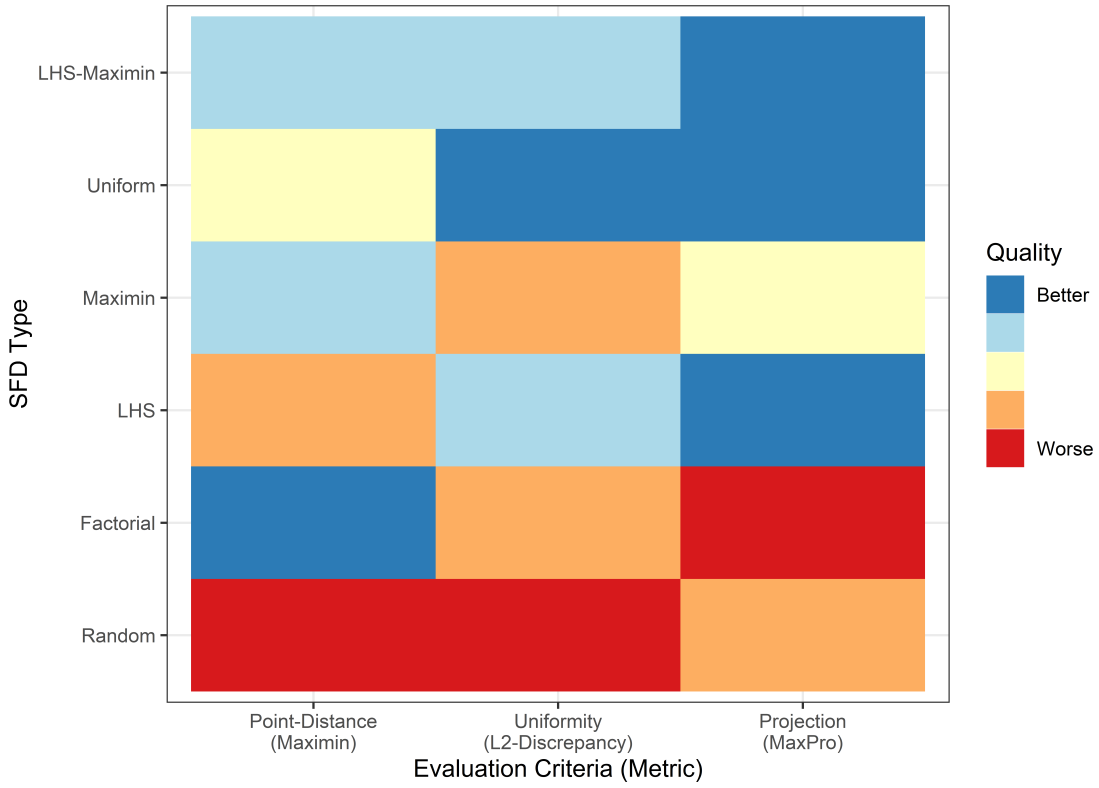


Figure 18. Summary of SFD performance across design evaluation criteria.

Red boxes indicate that a particular class of designs has relatively worse performance with respect to the associated property. Dark blue indicates the best relative performance in that category.

Recommendations

In formulating the recommendations summarized below, we considered performance across categories of evaluation criteria, implementability of the technique, and its validity in the literature. We also considered the consistency of designs produced using a given technique, because a smaller variation among iterations is preferable. Which technique is recommended depends on both the nature of the M&S input variables and the shape of the design space.

- **Continuous inputs only and no constrained regions** In this simplest case, there are many reasonable options. Of the options presented above, we would recommended LHS-Maximin if spacing out points is of key importance, and a Uniform design if uniform coverage is most important. These designs are both relatively intuitive, widely available in software, well established in the literature, and contain little to no variation among design iterations.

- **Categorical factors** If the M&S tool has both continuous and categorical inputs, Sliced LHS and FFF are both good options. Those who work well in the R environment may want to choose Sliced LHS, which is implemented via an R package. Those who prefer a graphical work environment may prefer to use FFF, which is implemented in JMP. FFF can also natively handle disallowed combinations, which might be an additional draw in some use cases.
- **Disallowed combinations or constrained regions** In most cases where there are constraints in the design region we would recommend FFF designs, for the reasons articulated above. The exception is if your project could benefit from detailed characterization of the distribution of importance across the parameter space, in which case we recommend using Minimum Energy.

Software Implementation

SFDs and associated analysis techniques are readily available in multiple software packages, including JMP and R.

JMP

Of the designs discussed above, JMP can build Uniform, Latin Hypercube, maximin (called Sphere Packing in JMP), Minimum Energy (called Minimum Potential in JMP), and FFF designs. Note that by default JMP provides the maximin LHS, thus preventing possible "bad" LHS designs discussed earlier. JMP can also fit polynomial and GP models to observed data.

R

Several R packages are useful for creating SFD:

- *lhs* - LHS designs
- *maximin* - Maximin designs
- *SLHD* - Sliced Latin Hypercube designs
- *mined* - Minimum Energy designs
- *DiceDesign* - Maximin and LHS designs, plus several evaluation criteria

Notably, FFF designs are not currently available in R. R also has packages for interpolators and GP modeling (*DiceKriging* or *RobustGASP* for continuous variables; *LVGP* for categorical and continuous variables).

References

Ba, Shan, William R. Myers, and William A. Brenneman. 2015. "Optimal Sliced Latin Hypercube Designs." *Technometrics* 57 (4): 479–87. https://doi.org/10.1080/00401706.2014.957867.

Chen, Victoria C. P., Kwok-Leung Tsui, Russell R. Barton, and Martin Meckesheimer. 2006. "A Review on Design, Modeling and Applications of Computer Experiments." *IIE Transactions* 38 (4): 273–91. https://doi.org/10.1080/07408170500232495.

Choudhuri, Nidhan, Subhashis Ghosal, and Anindya Roy. 2007. "Nonparametric Binary Regression Using a Gaussian Process Prior." *Statistical Methodology* 4: 227–43. https://doi.org/10.1016/j.stamet.2006.07.003.

Cioppa, Thomas M., and Thomas W. Lucas. 2007. "Efficient Nearly Orthogonal and Space-Filling Latin Hypercubes." *Technometrics* 49 (1): 45–55. https://doi.org/10.1198/004017006000000453.

National Research Council. 1998. *Statistics, Testing, and Defense Acquisition: New Approaches and Methodological Improvements*. National Academies Press.

Crombecq, K., E. Laermans, and T. Dhaene. 2011. "Efficient Space-Filling and Non-Collapsing Sequential Design Strategies for Simulation-Based Modeling." *European Journal of Operational Research* 214 (3): 683–96. https://doi.org/10.1016/j.ejor.2011.05.032.

Curlin, Carla, Toby Mitchell, Max Morris, and Don Ylvisaker. 1991. "Bayesian Prediction of Deterministic Functions, with Applications to the Design and Analysis of Computer Experiments." *Journal of the American Statistical Association* 86 (416): pp. 953–963.

Dalal, Ishaan L., Deian Stefan, and Jared Harwayne-Gidansky. 2008. "Low Discrepancy Sequences for Monte Carlo Simulations on Reconfigurable Platforms." In *International Conference on Application-Specific Systems, Architectures and Processors, 2008*. Piscataway, NJ: IEEE. https://doi.org/10.1109/asap.2008.4580163.

Damblin, G., M. Couplet, and B. Iooss. 2013. "Numerical Studies of Space-Filling Designs: Optimization of Latin Hypercube Samples and Subprojection Properties." *Journal of Simulation* 7 (4): 276–89. https://doi.org/10.1057/jos.2013.16.

Draguljić, Danel, Thomas J. Santner, and Angela M. Dean. 2012. "Noncollapsing Space-Filling Designs for Bounded Nonrectangular Regions." *Technometrics* 54 (2): 169–78.

Fang, Kai-Tai. 1980. "Uniform Design: Application of Number-Theoretic Methods in Experimental Design." *Acta Math. Appl. Sin.* 3: 363–72.

Fang, Kai-Tai, Dennis K. J. Lin, Peter Winker, and Yong Zhang. 2000. "Uniform Design: Theory and Application." *Technometrics* 42 (3): 237–48.

Garud, Sushant S., Iftekhar A. Karimi, and Markus Kraft. 2017. "Design of Computer Experiments: A Review." *Computers & Chemical Engineering* 106: 71–95. https://doi.org/10.1016/j.compchemeng.2017.05.010.

Gramacy, Robert B. 2020. *Surrogates: Gaussian Process Modeling, Design, and Optimization for the Applied Sciences*. CRC Press.

Hickernell, F. 1998. "Lattice Rules: How Well Do They Measure Up?" In *Random and Quasi-Random Point Sets*, edited by P. Hellekalek and G. Larcher, 1st ed., 106–66. Berlin/New-York: Springer-Verlag.

Husslage, Bart G. M., Gijs Rennen, Edwin R. van Dam, and Dick den Hertog. 2011. "Space-Filling Latin Hypercube Designs for Computer Experiments." *Optimization and Engineering* 12 (4): 611–30. https://doi.org/10.1007/s11081-010-9129-8.

Janssen, Hans. 2013. "Monte-Carlo Based Uncertainty Analysis: Sampling Efficiency and Sampling Convergence." *Reliability Engineering & System Safety* 109: 123–32. https://doi.org/10.1016/j.ress.2012.08.003.

Johnson, Mark E., Leslie M. Moore, and Donald Ylvisaker. 1990. "Minimax and Maximin Distance Designs." *Journal of Statistical Planning and Inference* 26 (2): 131–48.

Joseph, V. Roshan, Tirthankar Dasgupta, Rui Tuo, and C. F. Jeff Wu. 2015. "Sequential Exploration of Complex Surfaces Using Minimum Energy Designs." *Technometrics* 57 (1): 64–74. https://doi.org/10.1080/00401706.2014.881749.

Joseph, V. Roshan, Evren Gul, and Shan Ba. 2015. "Maximum Projection Designs for Computer Experiments." *Biometrika* 102 (2): 371–80. https://doi.org/10.1093/biomet/asv002.

Joseph, V. Roshan, Dianpeng Wang, Li Gu, Shiji Lyu, and Rui Tuo. 2019. "Deterministic Sampling of Expensive Posteriors Using Minimum Energy Designs." *Technometrics* 61 (3): 297–308. https://doi.org/10.1080/00401706.2018.1552203.

Lekivetz, Ryan, and Bradley Jones. 2015. "Fast Flexible Space-Filling Designs for Nonrectangular Regions." *Quality and Reliability Engineering International* 31 (5): 829–37. https://doi.org/10.1002/qre.1640.

Lekivetz, Ryan, and Bradley Jones. 2019. "Fast Flexible Space-Filling Designs with Nominal Factors for Nonrectangular Regions." *Quality and Reliability Engineering International* 35 (2): 677–84. https://doi.org/10.1002/qre.2429.

Liu, Haitao, Yew-Soon Ong, and Jianfei Cai. 2018. "A Survey of Adaptive Sampling for Global Metamodeling in Support of Simulation-Based Complex Engineering Design." *Structural and Multidisciplinary Optimization* 57 (1): 393–416. https://doi.org/10.1007/s00158-017-1739-8.

Loeppky, Jason L., Jerome Sacks, and William J. Welch. 2009. "Choosing the Sample Size of a Computer Experiment: A Practical Guide." *Technometrics* 51 (4): 366–76.

McKay, M. D., R. J. Beckman, and W. J. Conover. 1979. "A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code." *Technometrics* 21:

Mika, S., B. Schölkopf, A. J. Smola, K. R. Müller, M. Scholz, and G. Rätsch. 1998. "Kernel PCA and De-noising in Feature Spaces." *NIPS* 11: 536–42.

Montgomery, Douglas C. 2017. *Design and Analysis of Experiments*. John Wiley & Sons.

Picard, Rick, and Brian Williams. 2013. "Rare Event Estimation for Computer Models." *The American Statistician* 67 (1): 22–32. https://doi.org/10.1080/00031305.2012.751879.

Pronzato, Luc, and Werner G. Müller. 2012. "Design of Computer Experiments: Space Filling and Beyond." *Statistics and Computing* 22 (3): 681–701. https://doi.org/10.1007/s11222-011-9242-3.

Qian, Peter Z. G. 2012. "Sliced Latin Hypercube Designs." *Journal of the American Statistical Association* 107 (497): 393–99. https://doi.org/10.1080/01621459.2011.644132.

Sacks, J., W. J. Welch, T. J. Mitchell, and H. P. Winn. 1989. "Design and Analysis of Computer Experiments." *Statistical Science* 4 (2): 409–35.

Sacks, Jerome, Susannah B. Schiller, and William J. Welch. 1989. "Designs for Computer Experiments." *Technometrics* 31 (1): 41–47. https://www.jstor.org/stable/1270363.

Santner, Thomas J., Brian J. Williams, and William I. Notz. 2018. *The Design and Analysis of Computer Experiments*. 1st ed. New York City, New York, USA: Springer. https://doi.org/10.1007/978-1-4939-8847-1.

Wojton, Heather, Kelly Avery, Laura Freeman, Samuel Parry, Gregory Whittier, Thomas Johnson, and Andrew Flack. 2019. "Handbook on Statistical Design & Analysis Techniques for Modeling & Simulation Validation." Available at https://testscience.org/research-on-emerging-directions/

Wood, Simon N. 2017. *Generalized Additive Models; an Introduction with R*. 2nd ed. Boca Raton, Florida, United States of America: CRC Press.

Zemroch, Peter J. 1986. "Cluster Analysis as an Experimental Design Generator, with Application to Gasoline Blend Ing Experiments." *Technometrics* 28 (1): 39–49.

Zhang, Yichi, Siyu Tao, Wei Chen, and Daniel W. Apley. 2020. "A Latent Variable Approach to Gaussian Process Modeling with Qualitative and Quantitative Factors." *Technometrics* 62 (3): 291–302. https://doi.org/10.1080/00401706.2019.1638834.

Zhang, Yulei, and William I. Notz. 2015. "Computer Experiments with Qualitative and Quantitative Variables: A Review and Reexamination." *Quality Engineering* 27 (1): 2–13. https://doi.org/10.1080/08982112.2015.968039.

Zhu, Chong, Gopalakrishnan Sundaram, Jack Snoeyink, and Joseph SB Mitchell. 1996. "Generating Random Polygons with Given Vertices." *Computational Geometry* 6 (5): 277–90.

Appendix

This appendix provides additional details on specific evaluation criteria and metrics for SFD. Some of the content here provides additional mathematical background to the metrics discussed in the main text. We also included additional metrics not discussed in the main text. We use the following notation in multiple formulas that follow:

Notation	**Meaning**
n	The number of design points

Notation	Meaning
m	The number of factors under study
x_i	A design point, one of n
S_n	A design consisting of design points $x_1, \ldots, x_n$; a set
$d(x, y)$	The distance between points x and y (does not have to be Euclidean)
$y^{(p)}$	The p^{th} coordinate of the vector y
$\in$	Denotes set membership

Point-Distance

Point-distance metrics calculate the distance across samples or points, providing partial insight into the extent to which the placement of samples is optimal.

Maximin

Maximin (also referred to as "mindist") calculates the minimum pairwise distance between samples of the design (see equation below). In the preceding section, we discussed how maximization of this criterion can be used as a design optimization criterion. As a general evaluation criterion, higher maximin values can suggest a better design. As with any other evaluation criterion, however, maximin should not be used on its own to evaluate a design, as it provides a limited picture. Maximin only quantifies the worst-case scenario. Other than to provide the minimum, the criterion supplies no further information about how far apart the other pairs of samples are spaced. Also, designs with samples placed close to the boundary can have higher maximin values, causing undersampling towards the center of the parameter space.

Expressed mathematically, this metric is:

$$\min_{1 \leq i < j \leq n} d(x_i, x_j)$$

Minimax

Unlike maximin, which characterizes the relationship among samples, minimax provides insight into how the samples are placed in relation to all points within the parameter space. More specifically, minimax calculates maximum distance between all arbitrary points in the parameter space and their closest sample points (see equation below). Low minimax values therefore indicate that for any point in the space, there will be a sample nearby. This is a desirable quality in a space-filling design, as the samples are likely to be better representative of all other non-sampled points across the entire space.

Let T be a collection of points from the factor space (perhaps the whole space). Expressed mathematically, this metric is:

$$\max_{y \in T} \min_{i \leq i \leq n} d(y, x_i)$$

Average Distance

The minimax criterion only quantifies what the farthest sample-point distance is in a given design. Average distance, on the other hand, quantifies the overall representativeness of all samples. The initial step is the same as minimax, wherein all arbitrary points that are the closest to each sample are identified. Then, instead of the maximum, the average value is calculated. Lower values indicate more desirable designs.

Let $N(x_i)$ be the set of points in the collection T of points under consideration that are closer to x_i than any other design point. Below is a mathematical description of the metric:

$$\sum_{i=1}^{n} \sum_{y \in N(x_i)} \frac{d(x_i, y)}{|N(x_i)|}$$

Energy

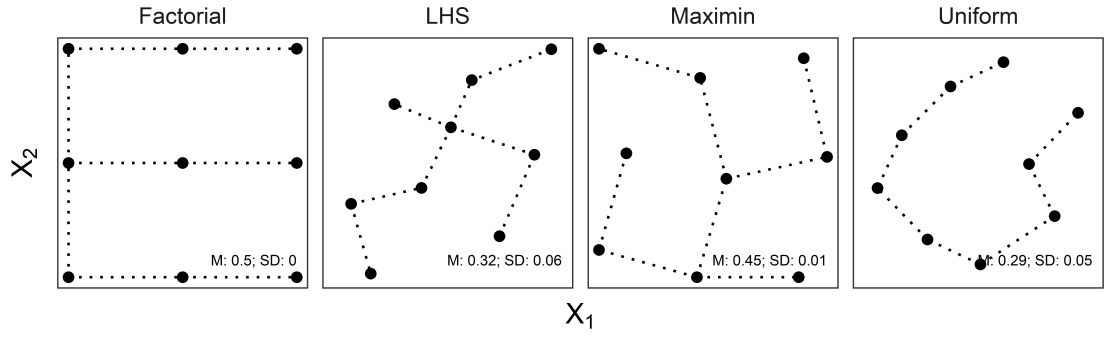
The energy metric quantifies the overall spread of samples across the parameter space using a concept from physics (Joseph et al. 2015). As discussed above as the optimization criterion for the Minimum Energy design, this metric calculates the potential energy of the space, treating each sample as an electric charge of equal sign. Here, lower energy values indicate more desirable designs. What makes this metric stand out from other point-distance criteria is its ability to assign charge values across the parameter space, thus effectively characterizing the value of over- or under-sampling a particular region.

If q represents an energy function (with constant energy corresponding to $q(x) = 1$) and k is a number greater than 1, the metric is expressed mathematically as:

$$\left(\sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \left(\frac{q(x_i)q(x_j)}{d(x_i, x_j)} \right)^k \right)^{1/k}$$

Minimum Spanning Tree

Minimum spanning tree (MST) is an approach that evaluates two properties: point-distance and uniformity of the design. MST refers to a tree-like structure that connects all the samples of the design without forming any "cycles" (i.e., no more than a single line can connect two points), keeping the overall length of lines as short as possible. Intuitively, this is similar to coming up with the cheapest way to connect all houses with power lines. Once MST is calculated, the mean length of the lines quantifies the overall point-distance, while the standard deviation shows how uniformly spaced apart the samples are. In general, a higher mean length indicates better point-distance and a lower standard deviation indicates better uniformity. As shown in the figures below, we can use the mean and standard deviation of these MST line segments to understand how classes of designs perform with respect to point distance and uniformity.



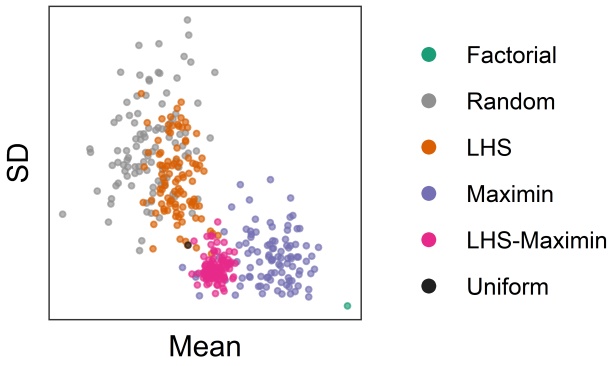


Figure A-1 - Comparison of Different Designs Using Minimum Spanning Tree Approach

The top panel shows minimum spanning trees for examples of Uniform, LHS, and Maximin designs in two dimensions. The mean and standard deviation of the line segments are given in the lower right. The bottom panel shows five different design types (100 iterations for Random, LHS, LHS-Maximin, Maximin; 1 for General Factorial and 1 for Uniform) plotted in this two-dimensional space, defined by mean segment length (x-axis; higher is better) and standard deviation of segment length (y-axis; lower is better). All designs were generated using two variables and nine samples.

Uniformity

Uniformity refers to the property of equal representation of all regions in the parameter space. It can be quantified in different ways; below we present metrics for doing so

L^2-Discrepancy

One way to define uniformity is to assess the extent to which the number of sample points included in subspaces of a design are close to the subspaces' volume. If the two sets of values are not close, then the design exhibits discrepancy from uniformity. Thus, a perfectly uniform design will have zero discrepancy.

There are many variants of the discrepancy metric depending on how the summary value of all defined subspaces is calculated. Star-discrepancy calculates the maximum magnitude of the difference in proportion of sample points in any subspace of the design space and said subspace's volume (Damblin, Couplet, and Iooss 2013). In practice, however, calculating the maximum of all possible subspaces is computationally intensive.

A popular alternative is the L^2-discrepancy, which is easier to compute as it uses the L^2 norm. Let $\{\cdots\}$ indicate a set (with the rule determining set contents written in the brackets) and $|\cdots|$ the number of set members (or the cardinality of the set) when applied to a set. The L^2-discrepancy can be written as:

$$\left(\int_{[0,1]^m} \left(\frac{1}{n} \left| \{x_i \in [0,y]\} \right| - \prod_{p=1}^{m} y^{(p)} \right)^2 \mathrm{dy} \right)^{1/2}$$

where the parameter space has been transformed to the hypercube $[0,1]^m$.[7] In the above equation, $[0,y]$ denotes the set of vectors with p^{th} coordinate between 0 and $y^{(p)}$ for all coordinates, and $[0,1]^m$ the set of vectors with all coordinates between 0 and 1 inclusively.

Many variants of the L^2-discrepancy exist. We recommend centered L^2-discrepancy, which can also help evaluate uniformity across multiple dimensions (Damblin, Couplet, and Iooss 2013).

[7] For the analytic form of the above equation, see Hickernell (1998).

Entropy

Entropy, applied to SFDs, is a measure of randomness of the distribution of samples across the parameter space. High entropy indicates that the samples are uniformly distributed across the space, which would be expected if they arose from a random distribution. Conversely, low entropy indicates that the samples are not randomly distributed, as when all samples are clumped in a corner, for instance.

See (Pronzato and Müller 2012) for details on computing this metric.

Projection

Not all variables involved in M&S may turn out to affect the response. It is thus desirable that samples of a space-filling design continue to provide coverage of the remaining variables, even if a subset will be discarded during analysis. This property is referred to projection. Poor projection indicates that if we collapse the design across one or more input variables, we have effectively wasted samples. While an ideal metric would assess projection properties for all possible subspaces defined by any combination of variables, such a metric would be extremely time-consuming to compute; in practice, we only consider limited combinations of variables (such as all pairs of variables, ignoring combinations of three).

MaxPro

The maximum projection (MaxPro) metric presented in (Joseph, Gul, and Ba 2015) seeks to encourage good coverage in all subspaces of a design space. Intuitively, an SFD with a good projection property will satisfy the following two conditions: (1) no two samples ever completely overlap in any individual dimension, and (2) all dimensions achieve a reasonable degree of uniformity. Because MaxPro is inversely scaled by the product of the square of all pairwise distances in each dimension, lower MaxPro values imply better projection properties. Thus, a design that fails to meet either of the two stated conditions will generate higher MaxPro values. For instance, any collapsing pair of samples will return the distance product as 0, causing MaxPro to reach infinity. In a less extreme example, lower pairwise distances in any of the dimensions will result in lower values for the distance product, driving the final MaxPro value to be higher.

This metric can also be used with categorical factors. Let $I\{A\}$ be an indicator function, which is equal to 1 if A is true and 0 otherwise. Let w_q be a non-negative weighting term and $c_i^{(q)}$ the q^{th} categorical factor's value for design point i out of h categorical factors. The following equation penalizes samples with overlapping categorical factors (Lekivetz and Jones 2019):

$$\left(\left(\frac{n}{2}\right)^{-1}\sum_{i=1}^{n-1}\sum_{j=i+1}^{n}\frac{\prod_{q=1}^{h}\left(1+w_q I\left\{c_i^{(q)}=c_j^{(q)}\right\}\right)}{\prod_{p=1}^{m}\left(x_i^{(p)}-x_j^{(p)}\right)}\right)^{1/p}.$$

Orthogonality

Here we include metrics that examine properties of the design matrix that could impact an analyst's ability to separate factor effects or obtain precise results.

Maximum Column Correlation

The primary goal in M&S is to understand how different variables influence the outcome response. It is therefore crucial to make sure that design samples are placed to allow dissociation of effects arising from those variables. For instance, in a design where the parameter values were perfectly correlated across two

variables, it would be impossible to attribute the finding to either of the variables even if the responses were to vary across the samples.

In more general terms, we want to make sure that the design is as orthogonal as possible. One way to assess the orthogonality of the design is to calculate the maximum magnitude of pairwise correlations across all variables. Lower values suggest higher orthogonality amongst variables, and thus better designs. Let ρ_{pq} be the correlation between factor settings for factors p and q in design S_n. A useful benchmark is 0.03 or lower for the equation below (Cioppa and Lucas 2007):

$$\max_{1 \leq p < q \leq m} |\rho_{\text{pq}}|$$

Conditioning Number

While a matrix's conditioning number is frequently used in numerical linear algebra applications, it can also reveal the degree of orthogonality of a proposed design matrix that has been centered and scaled to $-1, 1$. Large conditioning numbers indicate severe multicollinearity, that is, high correlation among input variables. A conditioning number of 1.13 or less is considered nearly orthogonal and free of serious multicollinearity concerns (Cioppa and Lucas 2007), though the minimum value of 1 can be obtained if the design matrix is orthogonal.

If ψ_1 is the largest singular value and ψ_m the smallest singular value of the design matrix, the conditioning number of the design matrix X is ψ_1/ψ_m. If ψ_m is zero, the conditioning number is infinite due to the matrix being rank-deficient.

Model-Specific Metrics

A popular tool for building M&S meta-models is GP regression, also known as kriging. Here we describe some of the metrics used to guide meta-model design generation. To those who are familiar with classical DOE and have used various "optimal" designs, this section may be familiar; because GPs are probabilistic models, you can make use of similar concepts from classical DOE. Thus, the metrics presented here are more closely related to the metrics of interest in classical DOE than the metrics from any of the previous sections, which seek only to describe some notion of how well points fill a factor space. None of the aforementioned metrics necessarily grant UQ guarantees.

Unfortunately, the equivalent metrics here are much more difficult to handle than in classical DOE; (Santner, Williams, and Notz 2018) notes they are rarely used by practitioners. The metrics depend greatly on the properties of the GP being fit to the data, including the unknown parameters (i.e., mean and covariance functions). Past experience may not be as helpful when building models for M&S systems; these parameters are not necessarily intuitive and can even be hard to guess. One approach for handling this issue is to use a model-free SFD to build an initial design, and then collect data with that design to build a preliminary model which you can employ to make the critical choices and estimates needed to use these metrics (Picard and Williams 2013). Another approach is to consider several possible values of the key parameters, form designs for each of them, then adopt whichever design is the most efficient (Jerome Sacks, Schiller, and Welch 1989). While this may provide some guidance on dealing with the uncertainty associated with key choices and parameters, the metrics unfortunately are difficult to compute, in addition to being difficult to optimize.[8]

Entropy

We discussed entropy earlier, presenting it as a metric describing a design's uniformity. We were referring to the entropy in the position of the points in the design space. Here we refer to the entropy in the associated

[8] As the metrics in this section are tailored to GPs, we assume GP knowledge in describing them. We briefly describe GPs in our section on analyzing M&S data, where we also provide references for further study.

models being fitted to the data. Otherwise, the notion of entropy remains; it is a measure of how "disordered" a system is. Here, though, entropy should be thought of as relating to information. Information is negatively related to entropy, and we want a design such that the fitted model has high information/low entropy. A design that is optimal in this sense is analogous to a D-optimal design from classical DOE (Curlin et al. 1991). Thus, while before we wanted the distribution of design points themselves to have high entropy, we want low entropy in the fitted model. Optimizing this metric directly can be difficult; it may be easier to generate a maximin-optimal design, which can be shown to be closely related to a D-optimal design under the right conditions (Johnson, Moore, and Ylvisaker 1990).

Here we give a mathematical description of what we would seek to optimize. If $\mathbf{R}$ is the correlation matrix of the GP associated with the design, then $-\det(\mathbf{R})$ is a measure of the amount of entropy in the fitted model (Santner, Williams, and Notz 2018), where $\det(\cdot)$ is the determinant of a matrix. We seek to minimize this metric, which is the same as maximizing $\det(R)$.

Integrated Mean Squared Prediction Error

The mean squared prediction error (MSPE) is the expected distance between the predicted and actual values of the response function at a site. If we are using a GP to interpolate the observed values of the response functions at design points, and the underlying process is deterministic, the predicted value of the response function at the design points will be the observed value, and the resulting MSPE at these sites will be zero. However, the MSPE at unobserved sites will not necessarily be zero. We would like MSPE to be "small," but as described, MSPE is not a single quantity, as it depends on the sites being considered. That said, we can construct metrics that measure how small MSPE tends to be.

One such metric is integrated MSPE (IMSPE). This metric "adds up" the MSPEs over the factor space. If IMSPE is small, MSPE overall tends to be small; hence, a good design should keep IMSPE small (J. Sacks et al. 1989). IMSPE-optimal designs correspond to A-optimal designs from classical DOE (Johnson, Moore, and Ylvisaker 1990); they also relate to L-optimal designs (Santner, Williams, and Notz 2018). This metric can be difficult to optimize directly, but optimal designs resemble maximin designs under the right conditions (Johnson, Moore, and Ylvisaker 1990).

See (Santner, Williams, and Notz 2018) for a mathematical description of this metric.

Maximum Mean Squared Prediction Error

One can alternatively describe how "small" the MSPE is by using its maximum value over the sample space; this is the maximum MSPE (MMSPE). Like IMSPE, we want MMSPE to be small. Unlike IMSPE, however, MMSPE grants a stronger guarantee, because the largest MSPE will not exceed the MMSPE. Thus the MMSPE is intuitively easier to understand. MMSPE-optimal designs correspond to G-optimal designs from classical DOE, and under certain conditions resemble minimax-optimal designs (Johnson, Moore, and Ylvisaker 1990).

See (Santner, Williams, and Notz 2018) for a mathematical description of this metric.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*	**2. REPORT TYPE**	**3. DATES COVERED** *(From - To)*
02-2021	IDA Publication	

4. TITLE ANDSUBTITLE

Space Filling Designs for Modeling & Simulation Validation

5a. CONTRACT NUMBER
HQ0034-19-D-0001

5b. GRANT NUMBER
____ ____ ____

5c. PROGRAM ELEMENT NUMBER
____ ____ ____

6. AUTHOR(S)

Kelly M. Avery (OED); Han G. Yi (OED); Curtis G. Miller (OED);

5d. PROJECT NUMBER
BD-09-2299

5e. TASK NUMBER
229990

5f. WORK UNIT NUMBER
____ ____ ____

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Institute for Defense Analyses
4850 Mark Center Drive
Alexandria, Virginia 22311-1882

8. PERFORMING ORGANIZATION REPORT NUMBER
NS-D-21562
H 2021-000048

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Director, Operational Test and Evaluation
The Pentagon
1700 Defense
Washington, D.C. 20301

10. SPONSOR/MONITOR'S ACRONYM(S)
DOT&E

11. SPONSOR/MONITOR'S REPORT NUMBER

12. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for Public Release. Distribution Unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT
System evaluations increasingly rely on modeling and simulation (M&S) to supplement live testing. Thus, it is crucial to thoroughly validate these M&S tools using rigorous data collection and analysis strategies. Space filling designs (SFD) are often the most effective and efficient way to collect data from the model and support a complete evaluation of the model's behavior. This paper is intended to be a survey of the SFD literature with the narrative specifically geared towards M&S validation and some of the unique challenges encountered in test & evaluation.

15. SUBJECT TERMS
Design of Experiments (DOE); Gaussian Process Modeling; Modeling & Simulation (M&S); Space Filling Designs; Statistics

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**			Heather Wojton (OED)
Unclassified	Unclassified	Unclassified	Unlimited	37	**19b. TELEPHONE NUMBER** *(include area code)* (703) 845-6811