Supplementary Online Zip File Instructions

Online ZIP file: This file contains Matlab code for implementing the sFFLHD methodology proposed in this paper. Extract all files in the zip file into a folder.

An example input orthogonal array is in the folder in a `.txt` file called `64run8level.txt`. The algorithm described in the paper is implemented in the Matlab function, `sFFLHD.m`.

With Matlab installed on your computer double click on the `sFFLHD.m` file. This will open Matlab and automatically use the working folder that contains the files: `64run8level.txt`, `sFFLHD.m`, and the supplementary function, `dec2bigbase.m`, that converts a decimal integer to a base B vector.

For those without access to Matlab, a Java implementation (by Joshua J. Burdett, LeTourneau University), that uses a graphical user interface to let you add batches singly or in groups, is available at **`http://harvest.nps.edu`**.

sFFHD function

Syntax

```
sFFLHD(OA, D, Max_N)
```

Description

`sFFLHD(OA, D, Max_N)` returns three matrices based on the orthogonal array, *OA*, each with *D* columns and *Max_N* rows. The three matrices are (1) the Big Grid Design **W**, (2) the Small Grid Design **V** and (3) the Design Matrix **X** for a sliced Full Factorial-Based Latin Hypercube Design as described in Duan, Ankenman, Sanchez, and Sanchez, *Technometrics* (2015).

The input *OA* must be an orthogonal array with *L* levels in each column designated by the integers 1 to *L*. *D* is the number of dimensions, and *Max_N* is (approximately) the maximum number of observations that you would potentially need. For calculation purposes, the smallest multiple of *L* greater than or equal to *Max_N* is used as the upper bound on the number of batches constructed. The design is to be implemented sequentially in practice, so it is presumed that stopping criteria prevent the number of observations from ever exceeding *Max_N*.

Since the *OA* has *L* levels in each column, the batch size for the sFFLHD will be *L*. The first *L* rows of each of the three matrices are the big grid design, the small grid design, and the design matrix for the first batch of observations, respectively. The other batches, all consisting of *L* rows, follow the same pattern.

Example

Create an sFFLHD sequential design that has 100 rows, batch size of 8 (L=8), and four dimensions (D=4) based on the orthogonal array in file **64run8level.txt**. Store the big grid design, the small grid design, and the design matrix in to the matrices, **V**, **W**, and **X**, respectively.

```
>> OA=csvread('64run8level.txt');
>> [V,W,X]= sFFLHD(OA, 4, 100);
```

(This will take a few minutes to complete. Notice that since 100 is not a multiple of 8, the matrices are of size 104×4.)