Hiking Mount Toblerone: Advanced Methods for Random Balance Design

Elmar Plischke, Stefano Tarantola, Thierry Mara
Outline

- Random Balance Design (RBD)
- De-biasing
- Random and quasi-random permutations (QRP)
- RBD with QRP
- Case study
Let us consider the input/output relation:

\[ Y = f(X_1, X_2, \ldots, X_k) \]

where:

- \( X_i \) are random variables
- \( f \) is a numerical simulation model
- \( Y \) is the model output
Basic set-up of RBD

1. Create a uniform sample $u \in [0,1]$ sampling from a periodic curve

$$u = \pi^{-1} \cos^{-1}(-\cos(\pi \omega (2s - 1)))$$

Where: $s \in [0,1]$

$\omega$ is an integer called basic frequency
Basic set-up of RBD

\[ u = \pi^{-1} \cos^{-1}(-\cos(\pi \omega (2s - 1))) \]

Example: \( \omega = 5; \)
Basic set-up of RBD

Most common case $\omega = 1$:

$$u = 1 - |2s - 1| \quad s \in [0,1]$$
In the original formulation:

\[ u_i(s) = \frac{1}{2} + \frac{1}{\pi} \arcsin(\sin \omega_i s) \quad s \in (-\pi; \pi) \quad \omega_i = 1 \quad i = 1, 2, ..., k \]
Basic set-up of RBD

\[ Y = f(X_1, X_2) = 3X_1 + 5X_2 \quad (X_1, X_2) \in [0;1]^2 \quad s \in (-\pi; \pi) \]

1) Sample N times from \(-\pi\) to \(\pi\) equidistantly (vector s) (ex N=8)

2) Feed s through the periodic curve: \( u(s) = \frac{1}{2} + \frac{1}{\pi} \arcsin(\sin \omega s) \)

3) Generate k indep. random permutations of u: \( p_i(u) = u_i \)

4) Transform \( u_i \) into \( x_i \) (not needed if iid \( U[0,1] \))

5) Evaluate model output \( y \quad y = f(x_1, x_2) \)

<table>
<thead>
<tr>
<th>s</th>
<th>u</th>
<th>x1</th>
<th>x2</th>
<th>y</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2.7489</td>
<td>0.375</td>
<td>0.625</td>
<td>0.875</td>
<td>6.25</td>
</tr>
<tr>
<td>-1.9635</td>
<td>0.125</td>
<td>0.125</td>
<td>0.625</td>
<td>3.50</td>
</tr>
<tr>
<td>-1.1781</td>
<td>0.125</td>
<td>0.875</td>
<td>0.125</td>
<td>3.25</td>
</tr>
<tr>
<td>-0.3927</td>
<td>0.375</td>
<td>0.375</td>
<td>0.125</td>
<td>1.75</td>
</tr>
<tr>
<td>0.3927</td>
<td>0.625</td>
<td>0.125</td>
<td>0.875</td>
<td>4.75</td>
</tr>
<tr>
<td>1.1781</td>
<td>0.875</td>
<td>0.875</td>
<td>0.375</td>
<td>4.50</td>
</tr>
<tr>
<td>1.9635</td>
<td>0.875</td>
<td>0.625</td>
<td>0.375</td>
<td>3.75</td>
</tr>
<tr>
<td>2.7489</td>
<td>0.625</td>
<td>0.375</td>
<td>0.625</td>
<td>4.25</td>
</tr>
</tbody>
</table>
Basic set-up of RBD

- Re-order $y$ by applying inverse permutations $p_1^{-1}$ and $p_2^{-1}$

Example with $N=128$
Basic set-up of RBD

- Looking for resonances of re-ordered $y$ at low frequencies

\[ \hat{S}_i = \frac{\sum_{l=1}^M |C_l|^2}{\sum_{l=1}^N |C_l|^2}, \quad i = 1, 2, \ldots, k \]

where $c_l$ are the Fourier coefficients of reordered $y$

$M = \text{max n. of harmonics (usually 4 or 6)}$

With $N$ points all $\hat{S}_i$ can be computed

$S_i = \frac{V_i}{V}$

crucial parameter
Basic set-up of RBD

\[ y = f(x_1, x_2) = 3x_1 + 5x_2 \]
Limitations of RBD

1. Sensitivity indices (especially small indices) are biased with respect to analytical values

2. Sample design in use does not necessarily cover the sample space uniformly. Estimates are affected by large random error

Improvements of RBD can be achieved by controlling these two drawbacks.
Bias in RBD

The factors $X_{-i}$ are randomly sampled.

The remaining part of variance $V_{-i}$ appears at all frequencies as random noise. A fraction of this noise overlaps to the signal at the lower harmonics.

Tissot and Prieur (2012) propose a bias correction formula based on the assumption that the unexplained variance $V_{-i}$ has a white noise

\[ S_{i}^{DB} = \frac{N\hat{S}_{i} - 2M}{N - 2M} \]

We will see a case study later
Large random error

2. Sample design in use does not necessarily cover the sample space with good uniform properties

Instead of random permutations, use permutations obtained from low-discrepancy sequences

\[
\begin{array}{c|cc|c|cc}
 s & p1 & p2 & u1 & u2 \\
-2.7489 & 1 & 1 & -2.7489 & -2.7489 \\
-1.9635 & 5 & 5 & 0.3927 & 0.3927 \\
-1.1781 & 3 & 7 & -1.1781 & 1.9635 \\
-0.3927 & 7 & 3 & 1.9635 & -1.1781 \\
0.3927 & 2 & 6 & -1.9635 & 1.1781 \\
1.1781 & 6 & 2 & 1.1781 & -1.9635 \\
1.9635 & 4 & 4 & -0.3927 & -0.3927 \\
2.7489 & 8 & 8 & 2.7489 & 2.7489 \\
\end{array}
\]

quasi-random balance design
Quasi-Random Balance Design

\[ y = \sum_{i=1}^{6} X_i \quad X_i \sim U[0,1] \]

We tested available Sobol’ sequence generators:

GNU scientific Library
Numerical Recipes
MatLab
Joe and Kuo
Broda Ltd.

\[ N = 512 \]

<table>
<thead>
<tr>
<th>Quasi-Random Source</th>
<th>( S_1 )</th>
<th>( S_2 )</th>
<th>( S_3 )</th>
<th>( S_4 )</th>
<th>( S_5 )</th>
<th>( S_6 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Analytical Values</td>
<td>0.1667</td>
<td>0.1667</td>
<td>0.1667</td>
<td>0.1667</td>
<td>0.1667</td>
<td>0.1667</td>
</tr>
<tr>
<td>GSL</td>
<td>0.1621</td>
<td>0.1632</td>
<td>0.1623</td>
<td>0.1627</td>
<td>0.1624</td>
<td>0.1628</td>
</tr>
<tr>
<td>NR</td>
<td>0.1627</td>
<td>0.1664</td>
<td>0.1614</td>
<td>0.1618</td>
<td>0.1867</td>
<td>0.1765</td>
</tr>
<tr>
<td>MatLab</td>
<td>0.1689</td>
<td>0.1683</td>
<td>0.1679</td>
<td>0.1677</td>
<td>0.1663</td>
<td>0.1657</td>
</tr>
<tr>
<td>Joe&amp;Kuo</td>
<td>0.1630</td>
<td>0.1622</td>
<td>0.1624</td>
<td>0.1623</td>
<td>0.1624</td>
<td>0.1632</td>
</tr>
<tr>
<td>Broda</td>
<td>0.1640</td>
<td>0.1626</td>
<td>0.1621</td>
<td>0.1644</td>
<td>0.1623</td>
<td>0.1622</td>
</tr>
<tr>
<td>Simple Random</td>
<td>0.1760</td>
<td>0.2137</td>
<td>0.2622</td>
<td>0.1842</td>
<td>0.1860</td>
<td>0.1557</td>
</tr>
<tr>
<td>Simple Random</td>
<td>0.1832</td>
<td>0.2022</td>
<td>0.1595</td>
<td>0.1873</td>
<td>0.2164</td>
<td>0.1846</td>
</tr>
</tbody>
</table>
Test Case

Ishigami test function

\[ y = \sin X_1 + 7 \sin^2 X_2 + 0.1X_3^4 \sin X_1 \quad X_i \sim U[-\pi, \pi] \]

\[ S_1 = 0.3138 \]

Analytic main effects: \[ S_2 = 0.4424 \]
\[ S_3 = 0 \]
The diagram shows a scatter plot with the x-axis labeled as "Sample Size" ranging from 200 to 2000, and the y-axis labeled as "$s^1$" ranging from 0.1 to 1.0. The plot displays a collection of data points that appear to be randomly distributed around a horizontal line at $s^1 = 0.3$. The data points are indicated by red dots. Additionally, there is a text annotation "RBD" on the right side of the graph.
$X_3$

Sample Size

$S_3$

RBD
QRBD
Second Test Case: a discontinuous function

\[ y = \sum_{i} X_i - \frac{\gamma}{2}, \]

\[ \gamma = 1 \text{ if } \exists x_i : 0 < x_i < 1/2 \]
\[ \gamma = 2 \text{ if } \exists x_{i,j} : 0 < x_{i,j} < 1/2 \]
\[ \gamma = 3 \text{ if } \exists x_{i,j,l} : 0 < x_{i,j,l} < 1/2 \]

Matlab function:

```matlab
def model(x):
    return sum(x, 2) - sum(x < .5, 2) / 2;
```

But in ten dimensions
Conclusions

QRBD is considerably superior to RBD and RBD-(de biased)

Much less random error and much better convergence

QRBD does not need to be debiased further

Plischke, Tarantola, Mara
Advanced Sensitivity Methods using Random Balance Design
In preparation

Thank you!