Screening Experiments with Maximally Balanced Projections

Tim Kramer and Shankar Vaidyaraman
October 2019

Overview

- Design focus – screening and robustness studies
- Orthogonality, near-orthogonality and balanced projections
- Some examples
- Outline of design optimization strategies
- Benefits and drawbacks
- Summary

Screening Designs

- Many potential factors
- Few factors will likely have large impact
- Experimental setup may impose balance constraints (e.g. plates with fixed wells per plate)
- Some factors will necessarily have more than two levels (solvents, HPLC column types, equipment types,…)
- Prediction confidence increases if design projects into full-factorial experiment for active factors

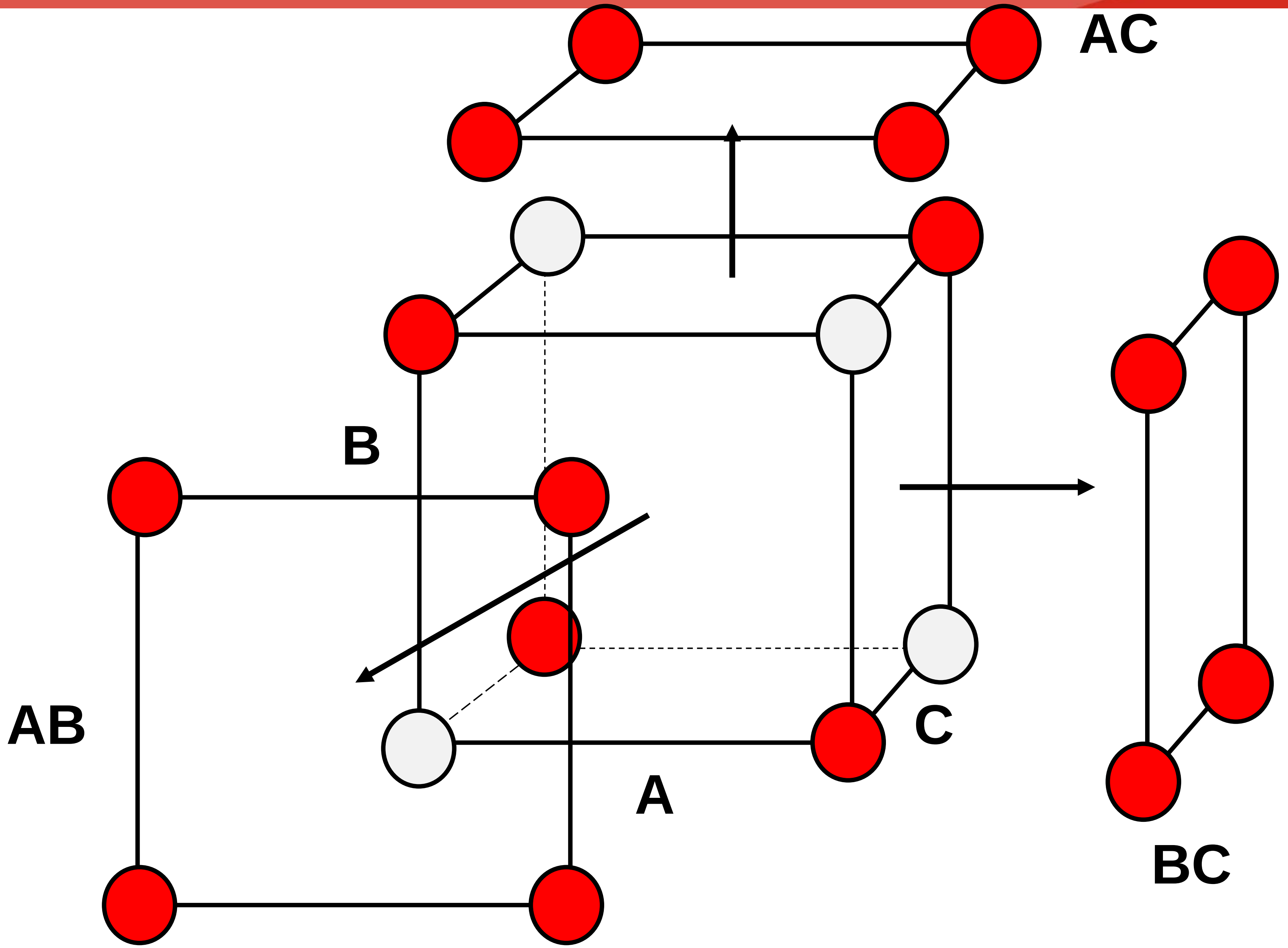
Robustness Designs

- Goal is not to identify impact of any specific factor
- Identify whether quality attribute is acceptable when an individual factor or combination of factors are at their extreme values
  - Want worst case combination of 1, 2, 3,… factors

Special Case Designs

- Calibration Designs: one key factor (e.g. concentration) in the presence of many noise factors
- Split Plot and Blocked Designs: Multiple factors applied to a set of plots or blocks

2-Dimensional Projections of Standard 2^3 Half-Fraction Design



Focus

- Previous work – examine projections of standard designs (fractional factorials, Plackett-Burman, Hadamard matrices, orthogonal arrays)
- Current work - reverse the process…create designs that have desired projection properties (for all combinations of two and three factors)

Some Previous Work

- Draper & Lin
- Box and Bisgaard
- Box and Tyssedal
- Tyssedal
- Hamada and Wu

Definitions of (Near) Orthogonality and (Balanced) Projectivity

- **Orthogonal Array (OA) of Strength *m*** (Rao, 1947): An array in which, for every *m*-tuple of columns, every level combination occurs *equally often* (m-balance)
- **Projectivity *m*** (Box and Tyssedal, 1996): An array in which, for every *m*-tuple of columns, every level combination occurs *at least once*
- **Nearly Orthogonal Array (NOA) of Strength *m*** (Lu, Li & Xie, 2006): *An array must have projectivity m and*, for every *m*-tuple of columns, every level combination must occur *as close to m-balance as possible*
- **Maximally Balanced Projectivity of Strength *m*** (Kramer): An array has MBC of strength *m* if, for every m-tuple of columns, every level combination occurs as close to m-balance as possible. (No "every level combination occurs at least once" requirement.)

Comparison of Projectivity Terms

N=18, Orthogonal Array of Strength 2

		f2		
		1	2	3
f1	1	2	2	2
	2	2	2	2
	3	2	2	2

N=12, Nearly Orthogonal Array of Strength 2

		f2		
		1	2	3
f1	1	1	1	2
	2	1	2	1
	3	2	1	1

N=12, Projectivity 2

		f2		
		1	2	3
f1	1	1	1	1
	2	1	4	1
	3	1	1	1

N=8, Maximally Balanced Projectivity of Strength 2

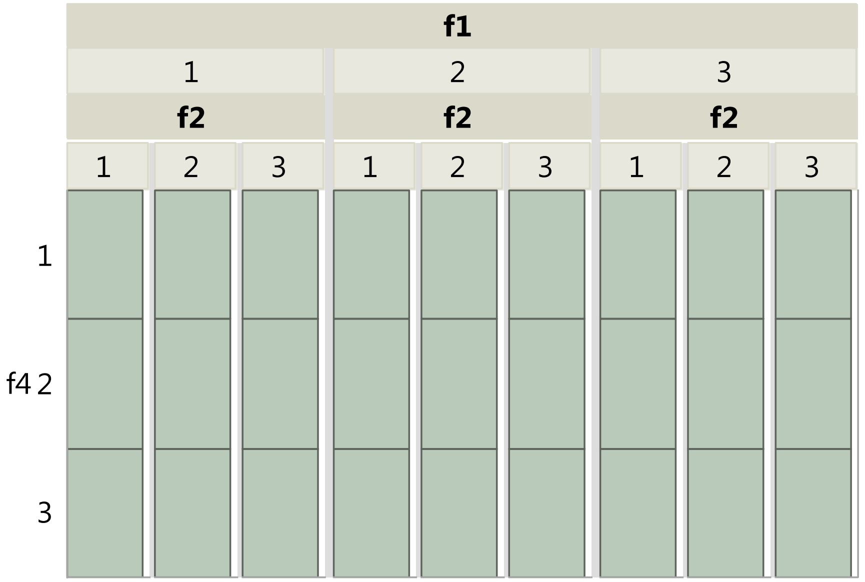
		f2		
		1	2	3
f1	1	1	1	1
	2	1	0	1
	3	1	1	1

Maximally Balanced Projections

- A projection is maximally balanced if the number of occurrences for each possible combination of a projection is as balanced as possible
- Example:
  - f1 (3 levels), f2 (3 levels), f3 (2 levels) in 12 runs
  - 18 combinations for f1*f2*f3 projection
  - Maximally balanced projection would have 12 ones and 6 zeroes. No preference to where the ones and zeroes are.

Balanced 3-Factor Projection: Three 3-Level Factors in 27 runs

	f1								
	1			2			3		
	f2			f2			f2		
f4	1	2	3	1	2	3	1	2	3
1	1	1	1	1	1	1	1	1	1
2	1	1	1	1	1	1	1	1	1
3	1	1	1	1	1	1	1	1	1

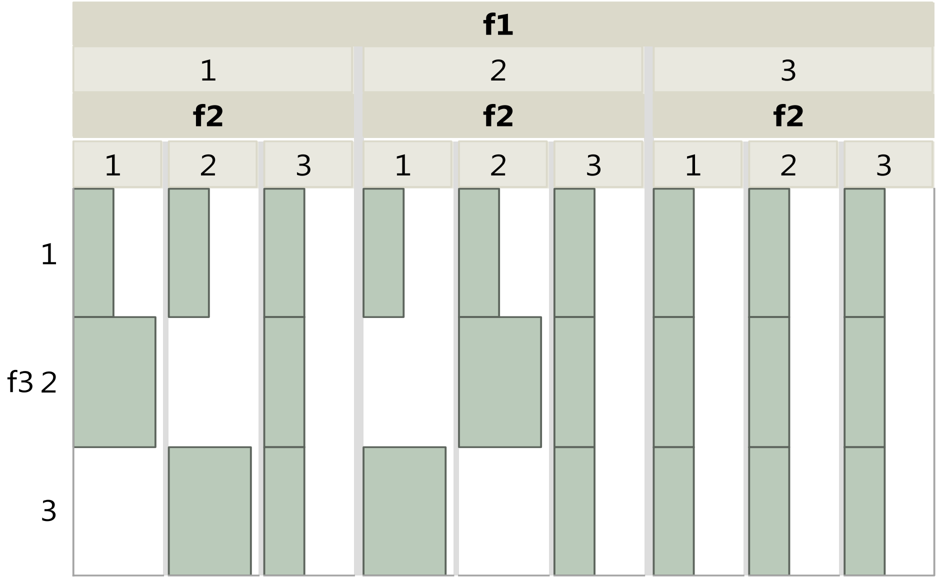


27 Cells

Maximally Balanced	1	1	…	1	27
Actual	1	1	…	1	27
Absolute Difference	0	0	0	0	0
Squared Difference	0	0	0	0	0

Imbalanced 3-Factor Projection: Three Level Factors in 27 Runs

	f1								
	1			2			3		
	f2			f2			f2		
f3	**1**	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**
1	1	1	1	1	1	1	1	1	1
2	2	0	1	0	2	1	1	1	1
3	0	2	1	2	0	1	1	1	1



	4 Cells				19 Cells				4 Cells				
Maximally Balanced	1	1	1	1	1	1	…	1	1	1	1	1	27
Actual	2	2	2	2	1	1	…	1	0	0	0	0	27
Absolute Difference	1	1	1	1	0	0	0	0	1	1	1	1	8
Squared Difference	1	1	1	1	0	0	0	0	1	1	1	1	8

Balanced 3 Factor Projections…but Different 1 and 2 Factor Projections For 3·3·2 Designs in 12 Runs

	F2					
	1		2		3	
	F3		F3		F3	
F1	1	2	1	2	1	2
1	1	0	1	1	0	1
2	1	1	0	1	1	0
3	0	1	1	0	1	1

	F2					
	1		2		3	
	F3		F3		F3	
F1	1	2	1	2	1	2
1	1	1	1	1	1	1
2	1	0	0	0	0	0
3	0	1	1	1	1	1

	F2					
	1		2		3	
	F3		F3		F3	
F1	1	2	1	2	1	2
1						
2						
3						

	F2					
	1		2		3	
	F3		F3		F3	
F1	1	2	1	2	1	2
1						
2						
3						

Imbalance Metrics for All Projections

Projection	Left Design = Maximally Balanced	Left Design Imbalance	Right Design Projection	Right Design Imbalance
F1	4, 4, 4	0	6, 5, 1	$2^2 + 1 + 3^2 = 14$
F2	4, 4, 4	0	4, 4, 4	0
F3	6, 6	0	6, 6	0
F1*F2	2, 2, 2, 1, 1, 1, 1, 1, 1	0	2, 2, 2, 2, 2, 1, 1, 0, 0	1 + 1 + 1 + 1 = 4
F1*F3	2, 2, 2, 2, 2, 2	0	3, 3, 3, 2, 1, 0	$1 + 1 + 1 + 1 + 2^2 = 8$
F2*F3	2, 2, 2, 2, 2, 2	0	2, 2, 2, 2, 2, 2	0
F1*F2*F3	12*1, 6*0	0	12*1, 6*0	0

Design Objective

- Identify designs having as balanced projections as possible:
  - One factor projections have highest priority…with no differentiation between individual factors
  - Two factor projections next highest priority
  - Three factor projections last priority
  - No four or higher factor projection evaluation

Brute Force Program to Identify Maximally Balanced Projection Designs (Exchange Algorithm)

- Start with cyclic assignment of levels for each factor: 1,2,…,k,1,2,…k,1,2,… for specified number of runs
- Randomize levels for each factor individually
- Randomize sequence of factors
- Evaluate imbalance (loss function)
- Re-order levels row by row, factor by factor trying to reduce imbalance
- Iterate until balanced-projections design is found or loss function seems as small as possible

Near Orthogonal Array Design with Mixed Integer Linear Programming (MILP)

Optimal Design with MILP

Expt	f0	f1	f2	f3
1	0	0	0	0
2	0	0	0	1
3	0	0	1	0
4	0	0	1	1
5	0	1	0	0
6	0	1	0	1
7	0	1	1	0
8	0	1	1	1
9	1	0	0	0
10	1	0	0	1
11	1	0	1	0
12	1	0	1	1
13	1	1	0	0
14	1	1	0	1
15	1	1	1	0
16	1	1	1	1

```
MILP:
Decision Variable: Which experiments are chosen
                   from full factorial design
Objective: Minimize imbalance
```

- Imbalance measure based on sum of absolute deviations
- Flexibility to add additional constraints such as maximum allowed deviation
- Challenge to solve larger instances due to lot of problem symmetry (future work to look into methods proposed in literature)

Bulutoglu, D. A., Margot, F., Classification of Orthogonal Arrays by Integer Programming, J. Stat. Planning and Inference, 138, 3, 654-666

Sequential Heuristic with MILP

Expt	f0	f1	f2	f3	f4
1			1	0	0
2			0	0	0
3			1	1	0
4			0	1	0
5			1	0	1
			0	0	1
			1	1	1
			0	1	1

Fixed partial design

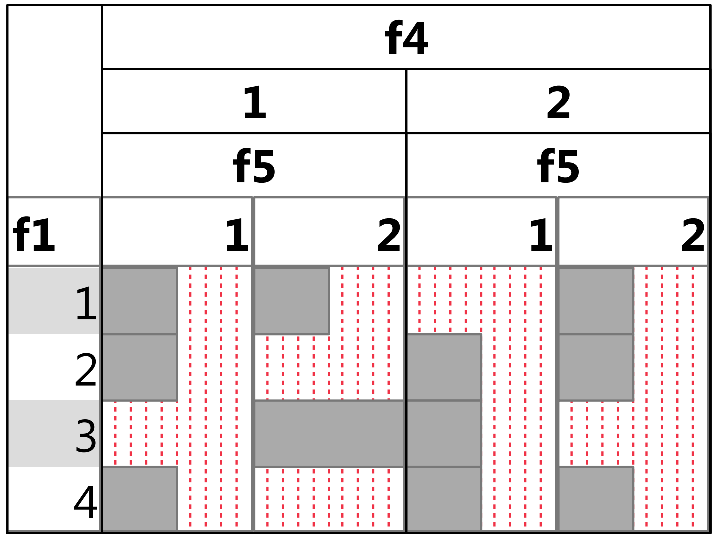
Design for a single factor based on MILP

```
Fix design for last factor N
For factor k from N-1 to 0:
    Solve MILP:
        Decision Variable: Levels of factor k in
                           each experiment
        Objective: Minimize imbalance of factor k
                   with the already determined design
                   for factors k+1:N
```

- Very fast to get a good design (based on empirical testing)
- Does not guarantee the best solution

Example 1: 4·3·3·2·2 in 12 runs

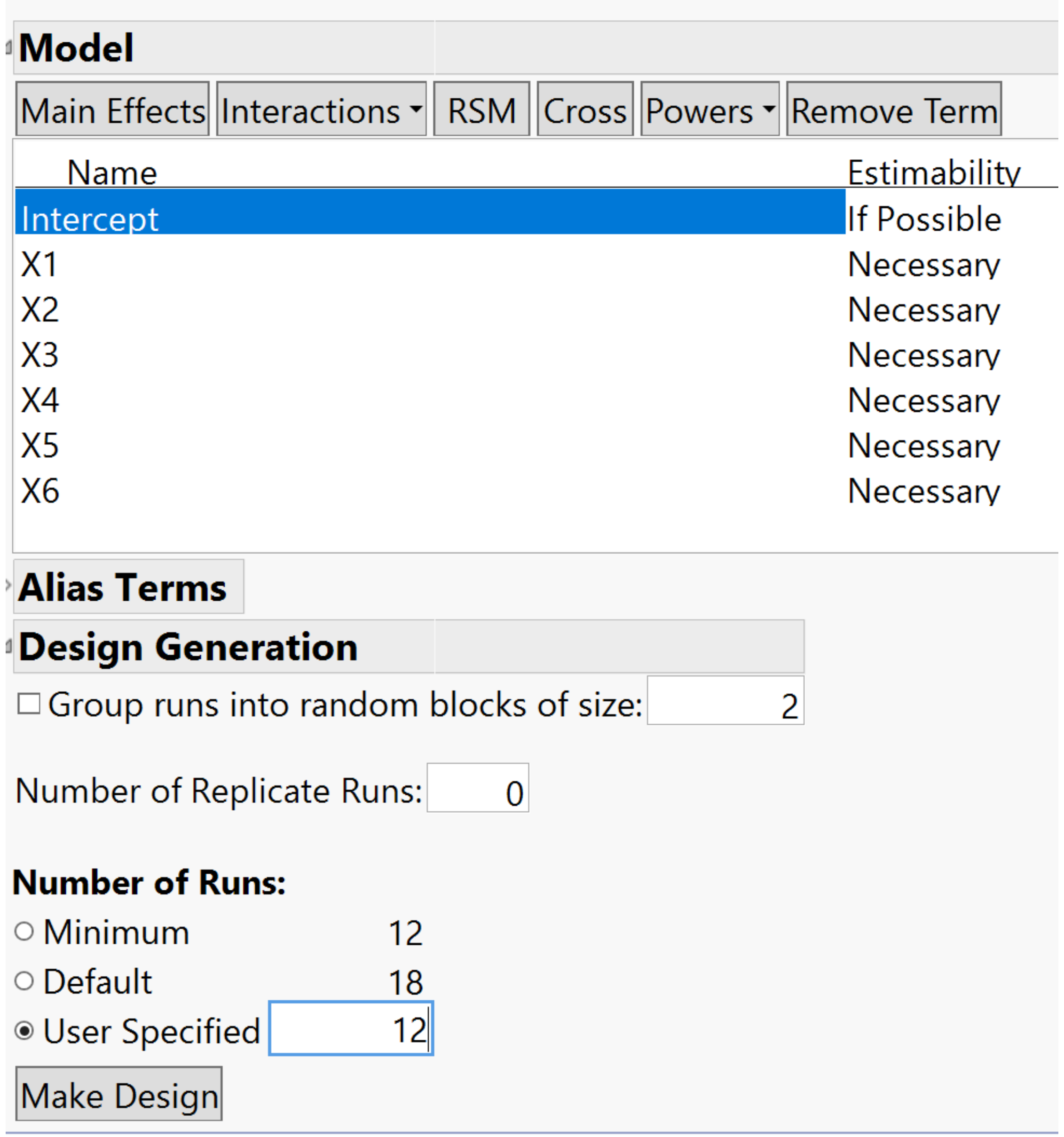
- D-optimal design readily obtained (D-efficiency of 96.268)
- Matches design obtained with maximally balanced projections
- Imbalance score of 2



f1	f2	f3	f4	f5
1	3	2	1	1
4	2	3	1	1
1	2	1	1	2
3	3	3	1	2
2	1	1	1	1
3	1	2	1	2
3	2	1	2	1
4	3	1	2	2
4	1	2	2	1
2	3	3	2	1
1	1	3	2	2
2	2	2	2	2

Example 2: 3·3·3·3·3·3 in 12 runs

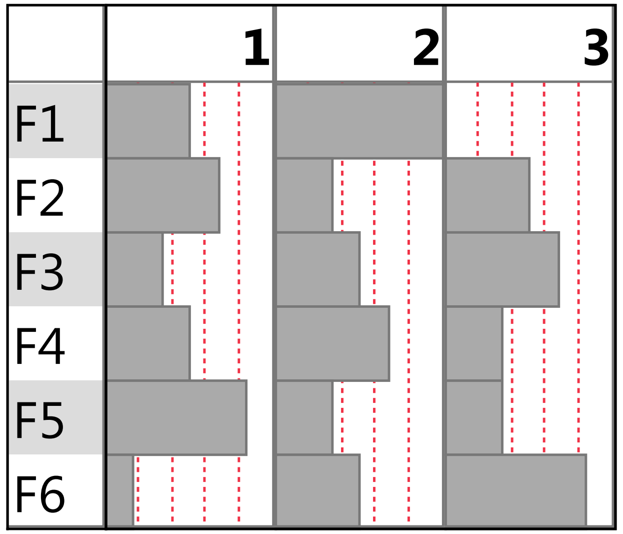
- No D-optimal design as 13 runs are necessary for intercept plus 2 effects per factor
- Can find a design if we designate estimation of intercept as "*if possible*"
  - Design will not generally be balanced
  - Uncertain what is being optimized



Example 2: 3·3·3·3·3·3 in 12 runs

F1	F2	F3	F4	F5	F6
2	3	3	2	3	2
2	3	2	2	1	1
2	2	1	3	3	3
1	3	3	3	1	3
2	3	1	1	2	3
1	1	2	1	3	3
1	1	1	2	1	2
1	2	3	2	2	3
2	1	2	3	2	2
2	2	2	1	1	2
3	1	3	2	1	3
2	1	3	1	1	1

	1	2	3
F1	4	7	1
F2	5	3	4
F3	3	4	5
F4	4	5	3
F5	6	3	3
F6	2	4	6



Projection	Dimensions	Imbalance
F1	1	18
F6	1	8
F5	1	6
F2	1	2
F3	1	2
F4	1	2
F1 * F6	2	10
F1 * F2	2	6
F1 * F3	2	6
F1 * F4	2	6
F1 * F5	2	6
F5 * F6	2	4
F3 * F6	2	4
F3 * F5	2	2
F4 * F5	2	2
F2 * F5	2	2
F2 * F6	2	2
F3 * F4	2	2
F4 * F6	2	2
F1 * F3 * F6	3	6
F1 * F5 * F6	3	2
F1 * F3 * F5	3	2
F3 * F5 * F6	3	2
F1 * F2 * F4	3	2
F1 * F4 * F5	3	2
F2 * F3 * F5	3	2
F2 * F4 * F5	3	2
F3 * F4 * F6	3	2

Example 2: 3·3·3·3·3·3 in 12 runs

F1	F2	F3	F4	F5	F6
3	2	2	1	3	1
1	2	3	2	3	3
3	3	1	2	3	2
2	3	3	3	3	1
1	1	2	3	3	2
3	1	3	2	1	1
1	2	1	3	1	1
2	2	2	2	1	2
1	3	3	1	1	2
2	1	1	1	2	3
3	2	3	3	2	2
1	3	2	2	2	1

With F6 instead of intercept designated as "If Possible"

	1	2	3
F1	5	3	4
F2	3	5	4
F3	3	4	5
F4	3	5	4
F5	4	3	5
F6	5	5	2

Projection	Dimensions	Imbalance
f6	1	6
f1	1	2
f2	1	2
f3	1	2
f4	1	2
f5	1	2
f3 * f6	2	2
f2 * f6	2	2
f4 * f6	2	2
f5 * f6	2	2
f1 * f6	2	2

Example 2: 3·3·3·3·3·3 in 12 runs

- Balanced design exists with balance in each one, two and three factor projection

	1	2	3
f1	4	4	4
f2	4	4	4
f3	4	4	4
f4	4	4	4
f5	4	4	4
f6	4	4	4

F1	F2	F3	F4	F5	F6
1	1	2	3	3	2
1	2	2	2	1	3
3	3	1	1	3	3
2	2	1	3	2	3
3	1	3	3	2	1
2	3	3	2	3	2
3	2	2	1	2	2
1	2	3	1	3	1
2	3	2	3	1	1
3	1	1	2	1	2
1	3	1	2	2	1
2	1	3	1	1	3

3·3·3·3·3 in 27 runs – Design 1

run	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
f1	1	2	3	3	1	2	2	3	1	3	1	2	3	2	1	2	1	3	2	3	1	2	1	3	1	3	2
f2	3	2	1	2	1	3	1	3	2	1	3	2	2	3	1	1	2	3	2	1	3	3	1	2	2	3	1
f3	1	1	1	2	2	2	3	3	3	1	1	1	2	2	2	3	3	3	1	1	1	2	2	2	3	3	3
f4	1	1	1	1	1	1	1	1	1	2	2	2	2	2	2	2	2	2	3	3	3	3	3	3	3	3	3
f5	3	2	1	3	2	1	3	2	1	3	2	1	2	3	1	2	3	1	3	2	1	2	3	1	2	3	1

Projection	Dimensions	Projection Imbalance	Total Imbalance
f1 * f2 * f3	3	54	54

3·3·3·3·3 in 27 runs – Design 2

run	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
f1	1	2	2	3	1	2	1	2	1	1	2	2	3	3	3	3	1	1	3	2	1	2	3	3	3	1	2
f2	3	1	2	2	1	3	3	2	2	1	2	1	2	1	3	1	3	2	2	3	2	3	3	1	3	1	1
f3	1	3	1	2	3	2	3	2	1	1	3	2	3	2	3	1	2	2	1	3	3	1	1	3	2	2	1
f4	1	3	3	1	1	2	2	2	1	3	1	1	3	3	3	1	3	3	2	1	2	3	2	2	1	2	2
f5	1	1	3	3	3	1	2	2	2	2	1	2	2	3	1	1	3	1	1	3	3	2	3	2	2	1	3

Projection	Dimensions	Projection Imbalance	Total Imbalance
f1 * f3 * f4	3	18	18
f1 * f3 * f5	3	18	36
f3 * f4 * f5	3	18	54

3·3·3·3·3 in 27 runs – Design 3

run	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
f1	1	2	1	1	2	3	3	3	3	3	1	2	2	3	1	3	3	2	2	2	1	1	2	1	3	1	2
f2	1	1	3	1	3	3	1	1	2	2	1	3	2	3	2	3	1	1	2	3	3	3	1	2	2	2	2
f3	2	2	3	3	1	2	3	1	3	1	1	3	1	3	2	1	2	3	2	2	2	1	1	1	2	3	3
f4	2	2	2	3	3	2	1	2	2	3	1	2	2	3	3	1	3	3	3	1	1	3	1	2	1	1	1
f5	3	2	1	2	2	3	3	1	2	3	1	3	1	1	1	2	2	1	3	1	2	3	3	2	1	3	2

Projection	Dimensions	Projection Imbalance	Total Imbalance
f2 * f3 * f4	3	18	18
f2 * f3 * f5	3	12	30
f2 * f4 * f5	3	12	42
f3 * f4 * f5	3	12	54

3·3·3·3·3 in 27 runs – Design 4

run	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
f1	2	1	3	1	2	3	2	3	1	1	3	2	3	2	1	3	3	2	1	2	2	3	3	2	1	1	1
f2	2	3	2	3	1	2	2	1	2	2	1	3	3	3	3	3	3	2	1	3	1	1	2	1	2	1	1
f3	1	3	3	2	1	1	2	2	3	2	2	1	3	2	3	1	1	3	1	2	3	3	2	3	1	1	2
f4	2	1	1	1	1	3	1	2	2	3	1	3	2	2	3	2	1	1	3	3	2	3	3	3	2	1	2
f5	1	1	1	3	1	3	3	3	3	1	2	3	3	2	2	1	2	2	2	1	2	1	2	3	2	3	1

Projection	Dimensions	Projection Imbalance	Total Imbalance
f1 * f2 * f3	3	12	12
f1 * f2 * f4	3	12	24
f1 * f3 * f5	3	12	36
f1 * f4 * f5	3	12	48
f2 * f3 * f4	3	12	60
f3 * f4 * f5	3	12	72

Benefits of Maximally Balanced Projections Design Strategy

- Easy to find designs for unusual combination of factor levels
  - Lilly experiment utilized 4·3·3·2·2 in 12 runs
- Tends to find "standard" (fractional factorial, Plackett-Burman) designs when they are appropriate
- Design strategy is consistent when going from unsaturated to saturated to supersaturated structures

Benefits of Maximally Balanced Projections Design Strategy

- If only two factors are important, design will generally project to a “good” design in those 2 dimensions (usually full factorial + extra runs)
- Blocking can be easily accommodated by including an additional factor

Drawbacks of Maximally Balanced Projections Design Strategy

- Prioritizes "balance" over other design considerations/criteria
- Balance criteria can lead to poor performance as D/A optimal design
- Can get "difficult" correlation structure
  - Can be addressed (Box and Meyer; Hamada and Wu)
- Doesn't guarantee that the number of runs is sufficient to estimate key factors

Additional Notes

- Split-plot designs require two-stage optimization
- For calibration designs, some noise factors may be continuous
  - Can assign levels to minimize correlation structure of these factors with calibration factor and each other

References

- Box, G. E. P. and Bisgaard, S. (1993). What Can You Find Out from 12 Experimental Runs, *Quality Engineering*, 5**,** pp. 663-668.
- Box, G. E. P. and Tyssedal, J. (1996). “Projective Properties of Certain Orthogonal Arrays”. *Biometrika,* **83**, pp. 950-955.
- Box, G. E. P. and Meyer, R. D. (1993). “Finding the Active Factors in Fractionated Screening Experiments”. *Journal of Quality Technology,* **25**, pp. 94-105.
- Chipman, H., Hamada, M. and Wu, C. F. J. (1997). “A Bayesian Variable Selection Approach for Analyzing Designed Experiments with Complex Aliasing”. *Technometrics,* **39**, pp. 372-381.
- Goos, P. and Jones, B. *Optimal Design of Experiments: A Case Study Approach,* John Wiley & Sons, Ltd. 2011.
- Hamada, M. S. and Wu, C. F. J. (1992), “Analysis of Designed Experiments with Complex Aliasing”. *Journal of Quality Technology,* **24**, pp. 130-137.

References

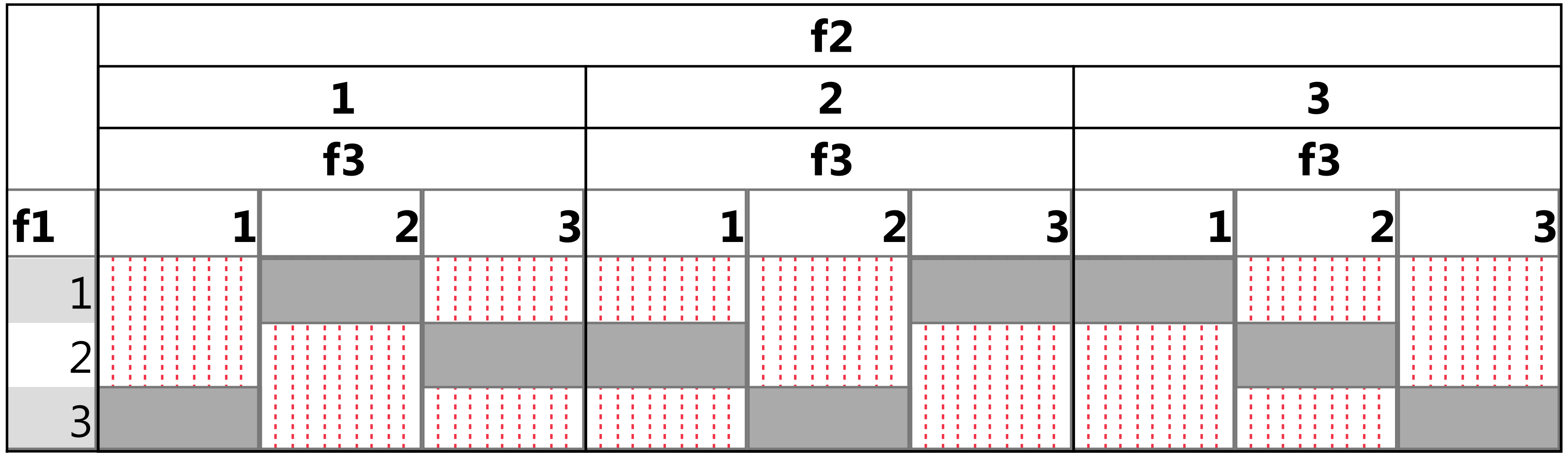
- Lin, D. K. J. and Draper, N. R. (1992). “Projection Properties of Plackett and Burman Designs”, *Technometrics*, **34**, pp. 423-428.
- Lu, X., Li, W. and Xie, M. (2006). “A Class of Nearly Orthogonal Arrays”. *Journal of Quality Technology,* **38**, pp. 148-161.
- Nguyen, N.-K. (1996). “A Note on the Construction of Near-Orthogonal Arrays with Mixed Levels and Economic Run Size”. *Technometrics,* **38**, pp. 279-283.
- Rao, C. R. (1947). “Factorial Experiments Derivable from Combinatorial Arrangements of Arrays”. *Journal of the Royal Statistical Society,* **9**, pp. 128-139.
- Samset, O. and Tyssedal, J. (1999). “Two-Level Designs with Good Projection Properties”, The Norwegian University of Science and Technology.
- Wang, J. C. and Wu, C. F. J. (1992). “Nearly Orthogonal Arrays with Mixed Levels and Small Runs”. *Technometrics,* **34**, pp. 409-422.

The End

3·3·3·3·3 in 27 runs – Design 1: Projection

	f2								
	1			2			3		
	f3			f3			f3		
f1	1	2	3	1	2	3	1	2	3
1	0	3	0	0	0	3	3	0	0
2	0	0	3	3	0	0	0	3	0
3	3	0	0	0	3	0	0	0	3

54



3·3·3·3·3 in 27 runs – Design 2: Projections

	f3								
	1			2			3		
	f4			f4			f4		
f1	**1**	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**
1	2	0	1	0	1	2	1	2	0
2	0	1	2	1	2	0	2	0	1
3	1	2	0	2	0	1	0	1	2

	f3								
	1			2			3		
	f5			f5			f5		
f1	**1**	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**
1	1	2	0	2	0	1	0	1	2
2	0	1	2	1	2	0	2	0	1
3	2	0	1	0	1	2	1	2	0

18 | 18
|
18 |

	f4								
	1			2			3		
	f5			f5			f5		
f3	**1**	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**
1	2	1	0	1	0	2	0	2	1
2	0	2	1	2	1	0	1	0	2
3	1	0	2	0	2	1	2	1	0

3·3·3·3·3 in 27 runs – Design 3: Projections

	f3								
	1			2			3		
	f4			f4			f4		
f2	**1**	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**
1	2	1	0	0	2	1	1	0	2
2	0	2	1	1	0	2	2	1	0
3	1	0	2	2	1	0	0	2	1

	f3								
	1			2			3		
	f5			f5			f5		
f2	**1**	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**
1	2	0	1	0	2	1	1	1	1
2	1	1	1	2	0	1	0	2	1
3	0	2	1	1	1	1	2	0	1

	f4								
	1			2			3		
	f5			f5			f5		
f2	**1**	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**
1	1	0	2	1	1	1	1	2	0
2	1	1	1	1	2	0	1	0	2
3	1	2	0	1	0	2	1	1	1

	f4								
	1			2			3		
	f5			f5			f5		
f3	**1**	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**
1	1	1	1	2	1	0	0	1	2
2	2	1	0	0	1	2	1	1	1
3	0	1	2	1	1	1	2	1	0

18 | 12
|
12 | 12

3·3·3·3·3 in 27 runs – Design 4: Projections

	f2								
	1			2			3		
	f3			f3			f3		
f1	**1**	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**
1	2	1	0	1	1	1	0	1	2
2	1	0	2	1	1	1	1	2	0
3	0	2	1	1	1	1	2	0	1

	f2								
	1			2			3		
	f4			f4			f4		
f1	**1**	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**
1	1	1	1	0	2	1	2	0	1
2	1	1	1	2	1	0	0	1	2
3	1	1	1	1	0	2	1	2	0

	f3								
	1			2			3		
	f5			f5			f5		
f1	**1**	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**
1	0	2	1	2	0	1	1	1	1
2	2	0	1	1	1	1	0	2	1
3	1	1	1	0	2	1	2	0	1

	f4								
	1			2			3		
	f5			f5			f5		
f1	**1**	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**
1	1	0	2	1	1	1	1	2	0
2	1	1	1	1	2	0	1	0	2
3	1	2	0	1	0	2	1	1	1

	f3								
	1			2			3		
	f4			f4			f4		
f2	**1**	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**
1	2	0	1	1	2	0	0	1	2
2	0	2	1	1	0	2	2	1	0
3	1	1	1	1	1	1	1	1	1

	f4								
	1			2			3		
	f5			f5			f5		
f3	**1**	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**
1	1	1	1	2	1	0	0	1	2
2	0	1	2	1	1	1	2	1	0
3	2	1	0	0	1	2	1	1	1

All 12’s