Supersaturated Designs: Introduction and New Research Results

1. Introduction (30 min): Definitions, Basic Ideas, and Related Literature

2. New Results (20 min): E(s^2)-Optimal Supersaturated Designs with Good Minimax Properties when N is Odd

Kenneth J. Ryan

kjryan@bgsu.edu

May 4, 2007

1. Introduction

- Basic Definitions
  - Two-Level Supersaturated Designs (SSDs)
  - E(s^2) Criterion
  - Minimax Criterion (s_{max}, $f_{s_{\text{max}}}$)
  - Isomorphic SSDs
- Lower Bounds for E(s^2) and Minimax
- A Search Algorithm for Finding Optimal SSDs
- Data Analysis

Definitions: SSDs

- A *two-level supersaturated design* (SSD) with N runs and m factors is a factorial design and is represented by an $N \times m$ matrix $\boldsymbol{X}$, where

  $N < m$,

  all entries of $\boldsymbol{X}$ are ± 1,

  the frequency of $+1$ in every column of $\boldsymbol{X}$ is $\lfloor N/2 \rfloor$, and

  no pair of columns of $\boldsymbol{X}$ have a dot product of $\pm N$.

This definition applies to odd or even N. However, for this first part of the talk, assume N is even unless stated otherwise.

Definitions (cont.): SSD Optimality Criteria Booth and Cox (1962)

- The E(s^2) value of SSD $\boldsymbol{X}$ is

$$\mathrm{E}(s^2) = \frac{\sum_{i<j} s_{ij}^2}{\binom{m}{2}},$$

where s_{ij} is the (i, j)th entry of $\boldsymbol{X}^T\boldsymbol{X}$.

- The *minimax criterion* orders SSDs by

$$s_{\max} = \max_{i<j} |s_{ij}|$$

and then by

$$f_{s_{\max}} = \sum_{i<j} \mathrm{I}_{s_{\max}}(|s_{ij}|).$$

Definitions (cont.): Isomorphic SSDs

- Two SSDs are *isomorphic* if one can be obtained from the other by permuting rows, permuting columns, or multiplying a subset of columns by -1.

- Isomorphic SSDs have the same E(s^2) value and minimax properties (s_{max}, $f_{s_{\text{max}}}$).

- For odd N, an SSD has one fewer $+1$ than -1 in each column. Odd N SSDs are thus isomorphic to designs where any subset of columns each have one more $+1$ than -1.

Lower Bounds for E(s^2)

A Goal: Find an SSD $\boldsymbol{X}$ with N runs and m factors having the lowest possible E(s^2), an E(s^2)-optimal SSD.

A Basic Concept: Derive the largest lower bound for E(s^2), so you can prove when you have an E(s^2)-optimal SSD $\boldsymbol{X}$.

- **Nguyen (1996) and Tang and Wu (1997):** Independently proved for even N that

$$\mathrm{E}(s^2) \geq \frac{m - N + 1}{(m-1)(N-1)} N^2.$$

- **Bulutoglu and Cheng (2004):** Derived an improved lower bound for E(s^2) which turned out to the largest lower bound for E(s^2) in many cases (N, m); see Ryan and Bulutoglu (2007).

Lower Bounds for Minimax

Another Goal: Find an SSD $\boldsymbol{X}$ with N runs and m factors having the lowest possible s_{max}. Over SSDs with lowest possible s_{max}, choose one with the lowest possible $f_{s_{\text{max}}}$. In other words, choose a *minimax optimal* SSD.

- **Lin (1993)** $s_{ij} \equiv N \pmod 4$.

- **Ryan and Bulutoglu (2007) Theorem 4:** Let $\boldsymbol{X}$ be an N run, m factor E(s^2)-optimal SSD. If $s_{\text{max}} \in \{2, 4, 6\}$, then $\boldsymbol{X}$ is minimax optimal as well.

A Search Algorithm: NOA

Nguyen (1996) near orthogonal array (NOA) algorithm.

- Exchange algorithm that finds a local E(s^2)-optimal SSD from a balanced, random starting design.

- Use largest available lower bound to try to prove E(s^2)-optimality.

- To promote numerical efficiency, compute $\boldsymbol{X}^T\boldsymbol{X}$ below the diagonal and use $\boldsymbol{X}^T\boldsymbol{X}$ updating formulas by Nguyen (1996).

An Example Search: $N = 8$ and $m = 9$

Step 1: Compute the largest available lower bound and generate a random balanced starting design.

Bulutoglu and Cheng (2004) bound: $\text{E}(s^2) \geq 3.56$.

Random balanced starting design $\boldsymbol{X}^{(0)} =$

-1	1	-1	-1	-1	-1	1	-1	-1
1	1	1	-1	-1	1	1	1	1
-1	1	1	1	1	-1	-1	-1	-1
1	-1	1	1	1	-1	1	-1	1
1	-1	-1	1	-1	1	-1	1	1
1	-1	-1	-1	-1	1	-1	-1	1
-1	1	-1	1	1	1	1	1	-1
-1	-1	1	-1	1	-1	-1	1	-1

with $\text{E}(s^2) = 7.11$, $s_{\text{max}} = 8$, and $f_{s_{\text{max}}} = 1$.

An Example Search: $N = 8$ and $m = 9$

Step 2: Swap a $+1$, -1 pair in the first column of $\boldsymbol{X}^{(0)}$ which produces the greatest reduction in E(s^2).

$\boldsymbol{X}^{(1)} =$

-1	1	-1	-1	-1	-1	1	-1	-1
1	1	1	-1	-1	1	1	1	1
1	1	1	1	1	-1	-1	-1	-1
1	-1	1	1	1	-1	1	-1	1
-1	-1	-1	1	-1	1	-1	1	1
1	-1	-1	-1	-1	1	-1	-1	1
-1	1	-1	1	1	1	1	1	-1
-1	-1	1	-1	1	-1	-1	1	-1

with E(s^2) $= 5.33$, $s_{\text{max}} = 4$, and $f_{s_{\text{max}}} = 12$.

An Example Search: $N = 8$ and $m = 9$

Step 3: Swap a $+1$, -1 pair in the second column of $\boldsymbol{X}^{(1)}$ which produces the greatest reduction in $\mathrm{E}(s^2)$.

$\boldsymbol{X}^{(2)} =$

-1	1	-1	-1	-1	-1	1	-1	-1
1	1	1	-1	-1	1	1	1	1
1	1	1	1	1	-1	-1	-1	-1
1	-1	1	1	1	-1	1	-1	1
-1	1	-1	1	-1	1	-1	1	1
1	-1	-1	-1	-1	1	-1	-1	1
-1	-1	-1	1	1	1	1	1	-1
-1	-1	1	-1	1	-1	-1	1	-1

with $\mathrm{E}(s^2) = 4.89$, $s_{\mathrm{max}} = 4$, and $f_{s_{\mathrm{max}}} = 11$.

An Example Search: $N = 8$ and $m = 9$

Step 4: Swap a $+1$, -1 pair in the third column of $\boldsymbol{X}^{(2)}$ which produces the greatest reduction in $\mathrm{E}(s^2)$.

$\boldsymbol{X}^{(3)} =$

-1	1	-1	-1	-1	-1	1	-1	-1
1	1	1	-1	-1	1	1	1	1
1	1	-1	1	1	-1	-1	-1	-1
1	-1	1	1	1	-1	1	-1	1
-1	1	1	1	-1	1	-1	1	1
1	-1	-1	-1	-1	1	-1	-1	1
-1	-1	-1	1	1	1	1	1	-1
-1	-1	1	-1	1	-1	-1	1	-1

with $\mathrm{E}(s^2) = 4.44$, $s_{\mathrm{max}} = 4$, and $f_{s_{\mathrm{max}}} = 10$.

An Example Search: $N = 8$ and $m = 9$

Step 5: Swap a $+1$, -1 pair in the fourth column of $\boldsymbol{X}^{(3)}$ which produces the greatest reduction in $\mathrm{E}(s^2)$.

$\boldsymbol{X}^{(4)} =$

-1	1	-1	-1	-1	-1	1	-1	-1
1	1	1	-1	-1	1	1	1	1
1	1	-1	1	1	-1	-1	-1	-1
1	-1	1	1	1	-1	1	-1	1
-1	1	1	1	-1	1	-1	1	1
1	-1	-1	-1	-1	1	-1	-1	1
-1	-1	-1	1	1	1	1	1	-1
-1	-1	1	-1	1	-1	-1	1	-1

with $\mathrm{E}(s^2) = 4.44$, $s_{\mathrm{max}} = 4$, and $f_{s_{\mathrm{max}}} = 10$.

An Example Search: $N = 8$ and $m = 9$

Step 6: Swap a $+1$, -1 pair in the fifth column of $\boldsymbol{X}^{(4)}$ which produces the greatest reduction in $\mathrm{E}(s^2)$.

$\boldsymbol{X}^{(5)} =$

-1	1	-1	-1	-1	-1	1	-1	-1
1	1	1	-1	1	1	1	1	1
1	1	-1	1	1	-1	-1	-1	-1
1	-1	1	1	-1	-1	1	-1	1
-1	1	1	1	-1	1	-1	1	1
1	-1	-1	-1	-1	1	-1	-1	1
-1	-1	-1	1	1	1	1	1	-1
-1	-1	1	-1	1	-1	-1	1	-1

with $\mathrm{E}(s^2) = 3.56 =$ bound, $s_{\mathrm{max}} = 4$, and $f_{s_{\mathrm{max}}} = 8$. $\boldsymbol{X}^{(5)}$ is $\mathrm{E}(s^2)$-optimal and minimax optimal, so the algorithm terminates.

Algorithms and Bounds

Algorithms and bounds need to be developed in tandem to solve design problems. One can validate the other.

In the $N = 8$ and $m = 9$ example, the older bound of Nguyen (1996) and Tang and Wu (1997)

$$\mathrm{E}(s^2) \geq \frac{9-8+1}{(9-1)(8-1)} 8^2 = 2.29$$

could not prove the E(s^2)-optimality of $\boldsymbol{X}^{(5)}$.

Similarly, a poor algorithm may not be able to locate an E(s^2)-optimal SSD to help prove that the Bulutoglu and Cheng (2004) bound $\mathrm{E}(s^2) \geq 3.56$ is the largest lower bound.

Data Analysis

Much of the SSD literature is on the design problem (e.g., algorithms and bounds to find and prove optimality of SSDs).

There is literature on how one would analyze a data set from an SSD–an $N \times 1$ response vector $\boldsymbol{Y}$ collected from an N run SSD $\boldsymbol{X}$.

Gilmour (2006) reviews the SSD data analysis literature and addresses the questions of if and when SSDs are practical.

2. E(s^2)-Optimal Supersaturated Designs with Good Minimax Properties when N is Odd

- Lower Bounds for E(s^2) and Minimax

- SSD Search Algorithms and Results
  - Near Orthogonal Array p (NOA$_p$)
  - Circulant NOA$_p$ (CNOA$_p$)

Lower Bounds for E(s^2) and Minimax

- **Bulutoglu and Ryan (2008) Theorem 1:** Improved the Nguyen and Cheng (2006) odd N lower bounds for E(s^2); see the back of the last page in your packet.

- **Lin (1993)** $s_{ij} \equiv N \pmod{4}$.

- **Bulutoglu and Ryan (2008) Theorem 5:** Let $\boldsymbol{X}$ be an N run, m factor E(s^2)-optimal SSD. If $s_{\max} \in \{1, 2, 3, 4, 6\}$, then $\boldsymbol{X}$ is minimax optimal as well.

Search Algorithms: NOA$_p$

Ryan and Bulutoglu (2007) NOA$_p$ algorithms modify the criterion of the Nguyen (1996) NOA algorithm to

$$f^p(\boldsymbol{X}) = \sum_{i<j} |s_{ij}|^p$$

for some fixed $p \in \mathbb{R}$. (NOA and NOA$_2$ are the same algorithm.)

When $p > 2$, the f^p criteria are a compromise between E(s^2) and minimax.

We have found that $p = 4$ works well for finding E(s^2)-optimal SSDs with good minimax properties.

NOA$_p$ Search Results

Cases where E(s^2)-optimal SSDs were found using NOA$_2$ and NOA$_4$ algorithms and Bulutoglu and Ryan (2008) Theorem 1.

N	m
5	5-10
7	7-30, 33-35
9	9, 11-111, 113-115, 117, 124
11	11-13, 15-24, 26-34
13	13, 15-17, 20, 21, 24, 27, 28, 32
15	15-17, 20, 21, 24
17	17

Solved odd N cases: 179 in all. Of these 179 cases, 70 could **not** be solved using the bound from Nguyen and Cheng (2006).

$\mathbf{NOA}_p$ Search Results (cont.)

Cases where E(s^2)-optimal, minimax optimal SSDs were found using NOA_2 and NOA_4 algorithms and Bulutoglu and Ryan (2008) Theorems 1 and 5.

N	m
5	5-10
7	7, 9, 10, 12
9	9, 11, 12
11	11, 13, 15
13	13, 15
15	15
17	17

This is a subset of the cases from the last slide.

k-Circulant SSDs

- A k-circulant SSD

  $$\boldsymbol{X} = [\boldsymbol{x}_1\ \boldsymbol{x}_2 \cdots \boldsymbol{x}_k\ \sigma\boldsymbol{x}_1\ \sigma\boldsymbol{x}_2 \cdots \sigma\boldsymbol{x}_k \cdots \sigma^{N-1}\boldsymbol{x}_1\ \sigma^{N-1}\boldsymbol{x}_2 \cdots \sigma^{N-1}\boldsymbol{x}_k]$$

  can be defined by its first k columns $\boldsymbol{x}_j$ $(j = 1, 2, \ldots, k)$.

- All other columns are defined via the cyclic shift operator σ. For example, $\sigma^1\boldsymbol{x}_j$ moves the elements of $\boldsymbol{x}_j$ down a row and makes the last element of $\boldsymbol{x}_j$ the first element of $\sigma^1\boldsymbol{x}_j$.

- A k-circulant SSD $\boldsymbol{X}$ has $m = Nk$ factors.

Searching k-Circulant SSD Spaces

- Liu and Dean (2004) exhaustive searches (even N).

- Koukouvinos *et al.* (2006a) and Koukouvinos *et al.* (2006b) used genetic and simulated annealing search algorithms, respectively, (even N).

- Bulutoglu and Ryan (2008) used a circulant NOA$_p$ (CNOA$_p$) algorithm. Only needed to store and update $k\lfloor N/2 \rfloor + \binom{k}{2} N$ of the
$$\binom{m}{2} = \binom{Nk}{2} = k\binom{N}{2} + \binom{k}{2} N^2$$
s_{ij} below the diagonal of $\boldsymbol{X}^T\boldsymbol{X}$.

CNOA_4 Search Results

Cases where E(s^2)-optimal, SSDs were found using CNOA_4 algorithms:

N	k
11	2-17
13	$2t$, $t = 1, 2, \cdots, 12$
15	2-14
17	$2t$, $t = 1, 2, \cdots, 13$
19	2-12
21	$2t$, $t = 1, 2, \cdots, 10$
23	2-16
25	$2t$, $t = 1, 2, \cdots, 10$.

Result: Add an all +1s row to each corresponding E(s^2)-optimal SSD in the table above. The resulting SSDs with even N are all E(s^2)-optimal.

$CNOA_4$ Search Results (cont.)

Cases where E(s^2)-optimal, minimax optimal, k-circulant SSDs were found using $CNOA_4$ algorithms followed by adding an all $+1$s row.

N	k
12	2, 3, 5, 6
14	2, 4, 6, 8^*, 10^*
16	2, 4^*
18	2, 4, 6^*, 8^*
20	2
22	2, 4^*
26	2^*

Asterisks indicate cases not found in other literature.