SELC :
Sequential Elimination of Level Combinations by Means of Modified Genetic Algorithms

Revision submitted to Technometrics

Abhyuday Mandal

Ph.D. Candidate
School of Industrial and Systems Engineering
Georgia Institute of Technology

Joint research with C. F. Jeff Wu and Kjell Johnson

SELC: Sequential Elimination of Level Combinations by means of Modified Genetic Algorithms

Outline

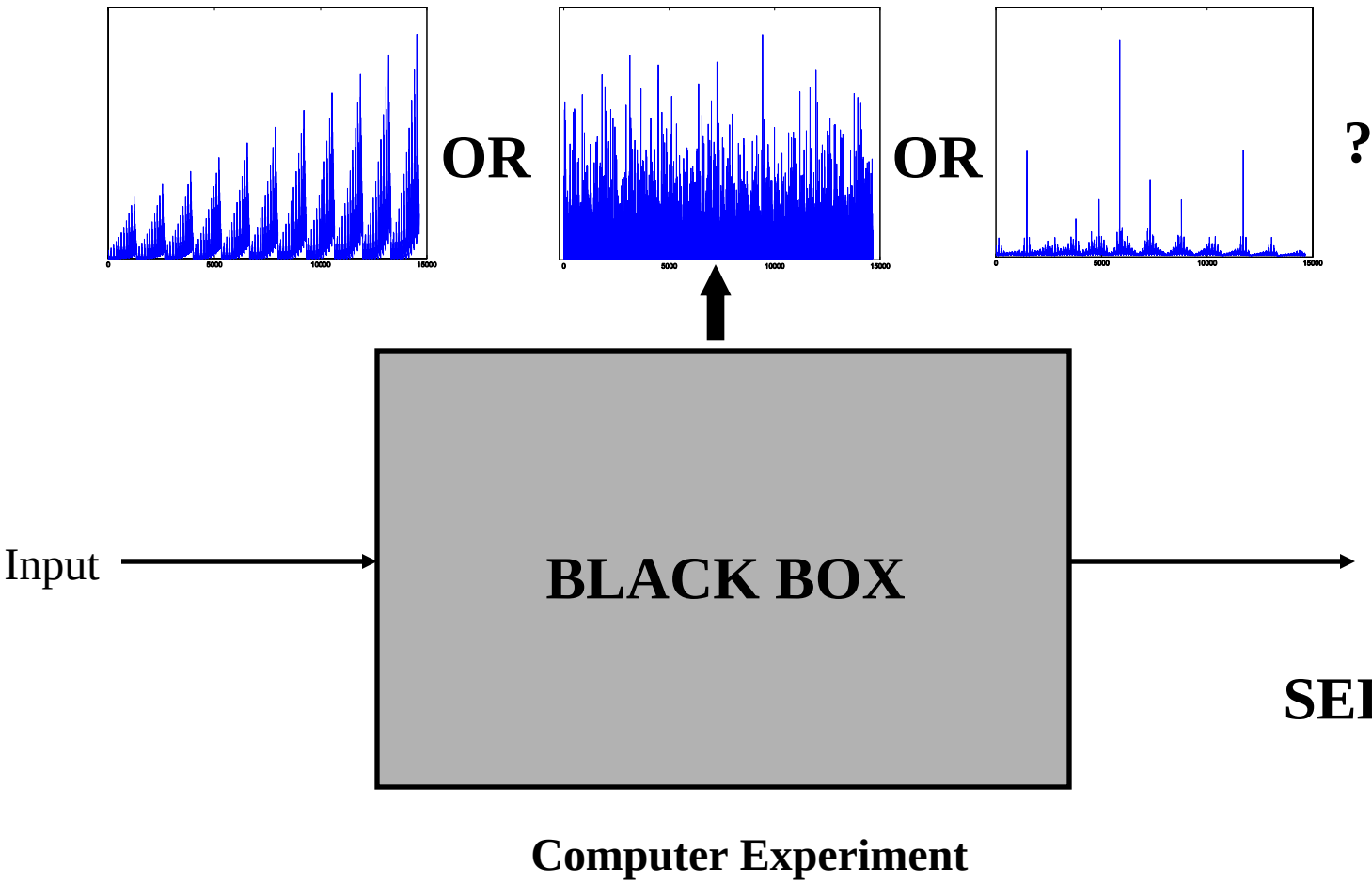
- Introduction – Motivational examples
- SELC Algorithm (Sequential Elimination of Level Combinations)
- Bayesian model selection
- Simulation Study
- Application
- Conclusions

What is SELC ?

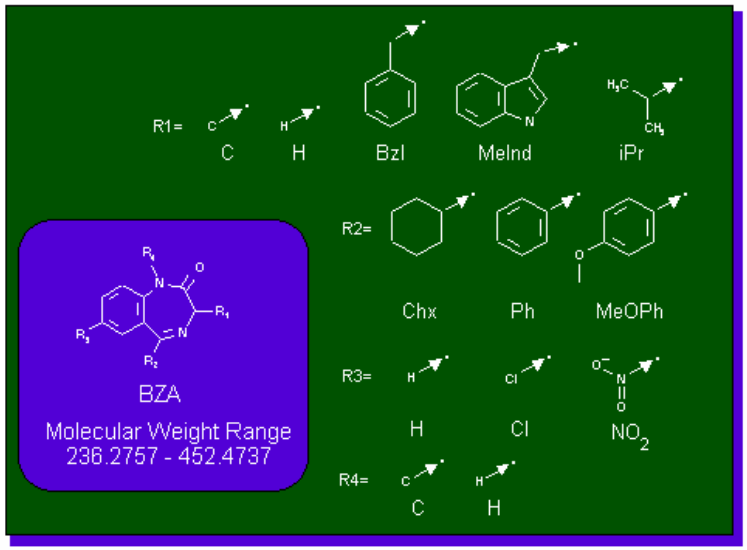
SELC = Sequential Elimination of Level Combinations

- SELC is a novel optimization technique which borrows ideas from statistics.
- Motivated by Genetic Algorithms (GA).
- A novel blending of Design of Experiment (DOE) ideas and GAs.
  - Forbidden Array.
  - Weighted Mutation (main power of SELC - from DOE.)
- This global optimization technique outperforms classical GA.

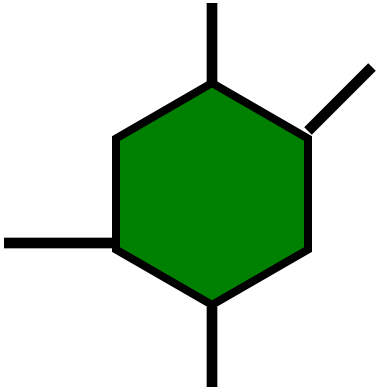
Motivating Examples



Example from Pharmaceutical Industry



$R_1, R_2, .., R_{10}$



$10 \times 10 \times 10 \times 10 = 10^4$ possibilities

SELC

Max

Sequential Elimination of Level Combinations (SELC)

A Hypothetical Example

$$y = 40 + 3A + 16B - 4B^2 - 5C + 6D - D^2 + 2AB - 3BD + \varepsilon$$

- 3 factors each at 3 levels.
- linear-quadratic system

level		*linear*	*quadratic*
1	$\longrightarrow$	-1	1
2		0	-2
3		1	1

- Aim is to find a setting for which y has maximum value.

Start with an OA($9, 3^4$)

A	B	C	D	y
1	1	1	1	10.07
1	2	2	3	53.62
1	3	3	2	43.84
2	1	2	2	13.40
2	2	3	1	46.99
2	3	1	3	55.10
3	1	3	3	5.70
3	2	1	2	43.65
3	3	2	1	47.01

Construct *Forbidden Array*

Forbidden Array is one of the key features of SELC algorithm.

First we choose the “worst” combination.

A	B	C	D	y
3	1	3	3	5.70

Forbidden array consists of runs with same level combinations as that of the “worst” one at any two positions:

A	B	C	D
3	1	*	*
3	*	3	*
3	*	*	3
*	1	3	*
*	1	*	3
*	*	3	3

where * is the wildcard which stands for any admissible value.

Order of Forbidden Array

- The number of level combinations that are prohibited from subsequent experiments defines the forbidden array's **order (*k*)**.

  – The lower the order, the higher the forbiddance.

Search for new runs

- After constructing the forbidden array, SELC starts searching for better level settings.
- **The search procedure is motivated by Genetic Algorithms.**

Search for new runs : Reproduction

- The runs are looked upon as chromosomes of GA.
- Unlike GA, binary representation of the chromosomes are not needed.
- Pick up the best two runs which are denoted by P_1 and P_2.

	A	B	C	D	y
P_1	2	3	1	3	55.10
P_2	1	2	2	3	53.62

- They will produce two offsprings called O_1 and O_2.

Pictorially

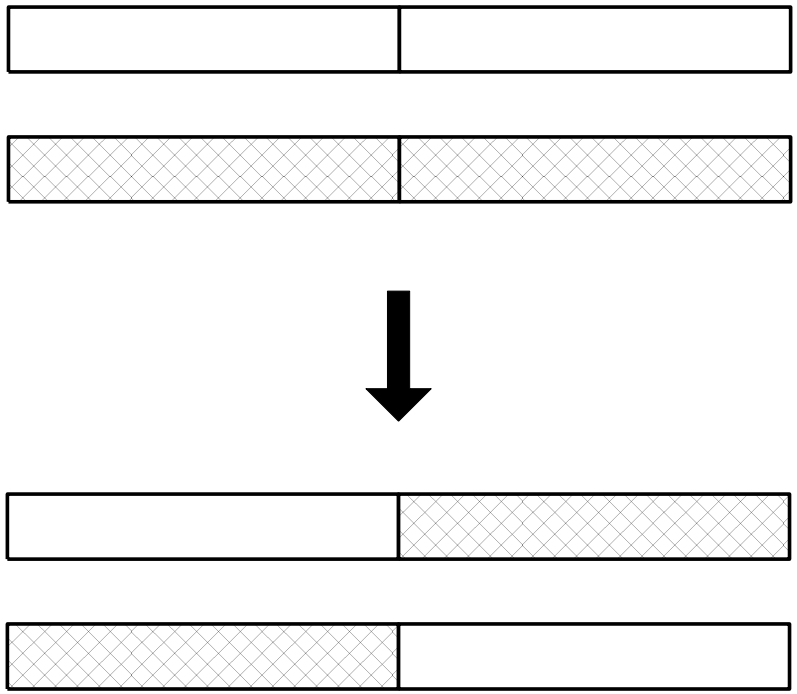


Figure 1 : Crossover

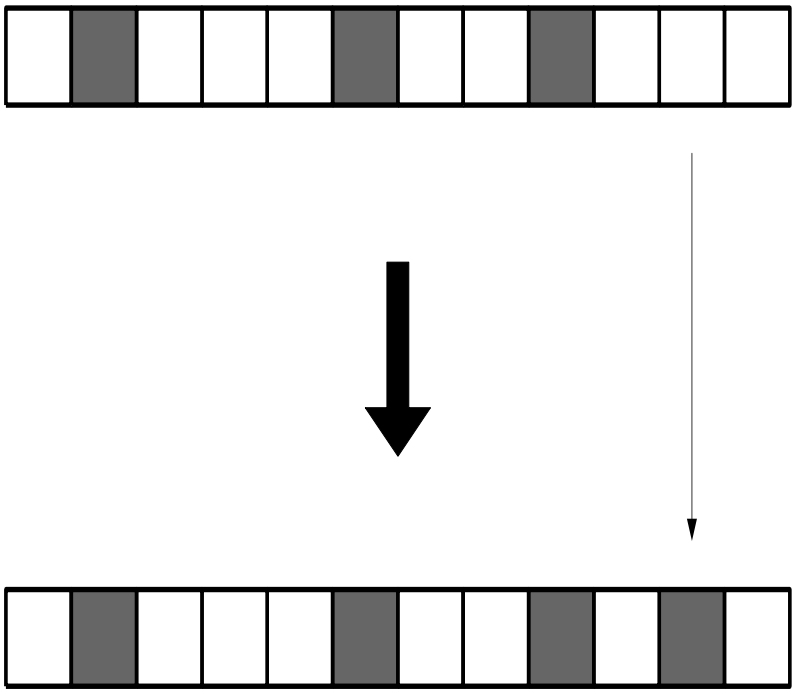


Figure 2 : Mutation

Step 1 – Crossover

Randomly select a location between 1 and 4 (say, 3) and do crossover at this position.

$$\begin{array}{lcccc|cc} P_1 & : & 2 & 3 & 1 & 3 \\ P_2 & : & 1 & 2 & 2 & 3 \end{array} \xrightarrow{Crossover} \begin{array}{lcccc|cc} O_1 & : & 2 & 3 & 2 & 3 \\ O_2 & : & 1 & 2 & 1 & 3 \end{array}$$

Step 2 – Weighted Mutation

Weighted Mutation is the driving force of SELC algorithm.

- **Design of Experiment ideas are used here to enhance the search power of Genetic Algorithms.**
- Randomly select a factor (gene) for O_1 and O_2 and change the level of that factor to any (not necessarily distinct) admissible level.
- If factor F has a significant main effect, then

$$\mathbf{p_l} \propto \bar{\mathbf{y}}(\mathbf{F} = \mathbf{l}).$$

- If factors F_1 and F_2 have a large interaction, then

$$\mathbf{q_{l_1 l_2}} \propto \bar{\mathbf{y}}(\mathbf{F_1} = \mathbf{l_1}, \mathbf{F_2} = \mathbf{l_2}).$$

- Otherwise the value is changed to any admissible levels.

Identification of important factors

Weighted mutation is done only for those few factors which are important (Effect sparsity principle).

A Bayesian variable selection strategy is employed in order to identify the significant effects.

Factor	*Posterior*
A	0.13
B	1.00
C	0.19
D	0.15
A^2	0.03
B^2	0.99
C^2	0.02
D^2	0.15

Factor	*Posterior*
AB	0.07
AC	0.03
AD	0.02
BC	0.06
BD	0.05
CD	0.03

Identification of important factors

If Factor B is randomly selected for mutation, then we calculate

$$p_1 = 0.09, p_2 = 0.45 \text{ and } p_3 = 0.46.$$

For O_1, location 1 is chosen and the level is changed from 2 to 1.

For O_2, location 2 was selected and the level was changed from 2 to 3.

$$\begin{array}{cccccccc} O_1 & : & 2 & 3 & 1 & 2 & \overset{Mutation}{\longrightarrow} & O_1 & : & \mathbf{1} & 3 & 1 & 2 \\ O_2 & : & 1 & 2 & 2 & 2 & & O_2 & : & 1 & \mathbf{3} & 2 & 2 \end{array}$$

Eligibility

An offspring is called ***eligible*** if it is not prohibited by the forbidden array.

Here both of the offsprings are eligible and are “new” level combinations.

A	B	C	D	y
1	1	2	1	10.07
1	2	1	2	53.62
1	3	3	3	43.84
2	1	1	1	13.40
2	2	3	3	46.99
2	3	2	2	55.10
3	1	3	1	5.70
3	2	2	1	43.65
3	2	3	2	47.01
1	3	1	2	54.82
1	3	2	2	49.67

Repeat the procedure

A	B	C	D	y
1	1	2	1	10.07
1	2	1	2	53.62
1	3	3	3	43.84
2	1	1	1	13.40
2	2	3	3	46.99
2	3	2	2	55.10
3	1	3	1	5.70
3	2	2	1	43.65
3	2	3	2	47.01
1	3	1	2	54.82
1	3	2	2	49.67
2	3	1	2	58.95
1	2	2	3	48.41
2	3	2	2	55.10
2	2	2	1	41.51
3	3	1	2	63.26

Stopping Rules

The stopping rule is subjective.

- As the runs are added one by one, the experimenter can decide, in a sequential manner, whether significant progress has been made and can stop after near optimal solution is attained.
- Sometimes, there is a target value and once that is attained, the search can be stopped.
- Most frequently, the number of experiments is limited by the resources at hands.

The SELC Algorithm

1. Initialize the design. Find an appropriate **orthogonal array**. Conduct the experiment.
2. Construct the **forbidden array**.
3. Generate new *offspring*.
   - *Select* offspring for reproduction with probability proportional to their "fitness."
   - *Crossover* the offspring.
   - *Mutate* the positions using **weighted mutation**.
4. Check the **new offspring's eligibility**. If the offspring is eligible, conduct the experiment and go to step 2. If the offspring is ineligible, then repeat step 3.

A Justification of Crossover and Weighted Mutation

Consider the problem of maximizing $K(\mathbf{x})$, $\mathbf{x} = (x_1, \ldots, x_p)$, over $a_i \leq x_i \leq b_i$.

Instead of solving the p-dimensional maximization problem

$$\max \left\{ K(\mathbf{x}) : a_i \leq x_i \leq b_i, i = 1, \ldots, p \right\}, \tag{1}$$

the following p one-dimensional maximization problems are considered,

$$\max \left\{ K_i(x_i) : a_i \leq x_i \leq b_i, i = 1, \ldots, p \right\}, \tag{2}$$

where $K_i(x_i)$ is the ith marginal function of $K(\mathbf{x})$,

$$K_i(x_i) = \int K(\mathbf{x}) \prod_{j \neq i} dx_j$$

and the integral is taken over the intervals $[a_j, b_j]$, $j \neq i$.

A Justification of Crossover and Weighted Mutation

Let x_i^* be a solution to the ith problem in (2). The combination $\mathbf{x}^* = (x_1^*, \ldots, x_p^*)$ may be proposed as an approximate solution to (1).

A sufficient condition for $\mathbf{x}^*$ to be a solution of (1) is that

$K(\mathbf{x})$ can be represented as

$$K(\mathbf{x}) \quad = \quad \psi\Big(K_1(x_1), \ldots, K_p(x_p)\Big) \tag{3}$$

and

ψ is nondecreasing in each K_i.

A special case of (3), which is of particular interest in statistics, is

$$K(\mathbf{x}) \quad = \quad \sum_{i=1}^{p} \alpha_i K_i(x_i) + \sum_{i=1}^{p} \sum_{j=1}^{p} \lambda_{ij} K_i(x_i) K_j(x_j).$$

SELC performs well in these situations.

Identification of Model : A Bayesian Approach

- Use Bayesian model selection to identify most likely models (Chipman, Hamada and Wu, 1997).
- Require prior distributions for the parameters in the model.
- Approach uses standard prior distributions for regression parameters and variance.
- Key idea : inclusion of a latent variable (δ) which identifies whether or not an effect is in the model.

Linear Model

For the linear regression with normal errors,

$$Y = X_i \beta_i + \varepsilon,$$

where

- Y is the vector of N responses,
- X_i is the ith model matrix of regressors,
- β_i is the vector of factorial effects (linear and quadratic main effects and linear-by-linear interaction effects) for the ith model,
- ε is the iid $N(0, \sigma^2)$ random errors

Prior for Models

Here the prior distribution on the model space is constructed via simplifying assumptions, such as independence of the activity of main effects (Box and Meyer 1986, 1993), and independence of the activity of higher order terms conditional on lower order terms (Chipman 1996, and Chipman, Hamada, and Wu 1997).

Let's illustrate this with an example. Let $\delta = (\delta_A, \delta_B, \delta_C, \delta_{AB}, \delta_{AC}, \delta_{BC})$

$$
\begin{aligned}
P(\delta) &= P(\delta_A, \delta_B, \delta_C, \delta_{AB}, \delta_{AC}, \delta_{BC}) \\
&= P(\delta_A, \delta_B, \delta_C) P(\delta_{AB}, \delta_{AC}, \delta_{BC} | \delta_A, \delta_B, \delta_C) \\
&= P(\delta_A) P(\delta_B) P(\delta_C) P(\delta_{AB} | \delta_A, \delta_B, \delta_C) P(\delta_{AC} | \delta_A, \delta_B, \delta_C) P(\delta_{BC} | \delta_A, \delta_B, \delta_C) \\
&= P(\delta_A) P(\delta_B) P(\delta_C) P(\delta_{AB} | \delta_A, \delta_B) P(\delta_{AC} | \delta_A, \delta_C) P(\delta_{BC} | \delta_B, \delta_C)
\end{aligned}
$$

Basic assumptions for Model selection

A1. *Effect Sparsity*: The number of important effects is relatively small.

A2. *Effect Hierarchy*: Lower order effects are more likely to be important than higher order effect and effects of the same order are equally important.

A3. *Effect Inheritance*: An interaction is more likely to be important if one or more of its parent factors are also important.

Prior for Distribution of Latent Variable δ

Main Effects

$$P(\delta_A = 1) = p$$

Quadratic Effects

$$P(\delta_{A^2} = 1|\delta_A) = \begin{cases} 0.1p & \text{if} \quad \delta_A = 0, \\ p & \text{if} \quad \delta_A = 1. \end{cases}$$

2fi's

$$P(\delta_{AB} = 1|\delta_A, \delta_B) = \begin{cases} 0.1p & \text{if} \quad \delta_A + \delta_B = 0, \\ 0.5p & \text{if} \quad \delta_A + \delta_B = 1, \\ p & \text{if} \quad \delta_A + \delta_B = 2. \end{cases}$$

The posterior probabilities of $\beta's$ are computed using Gibbs sampler.

Example 1 : Shekel 4 function (SQRIN)

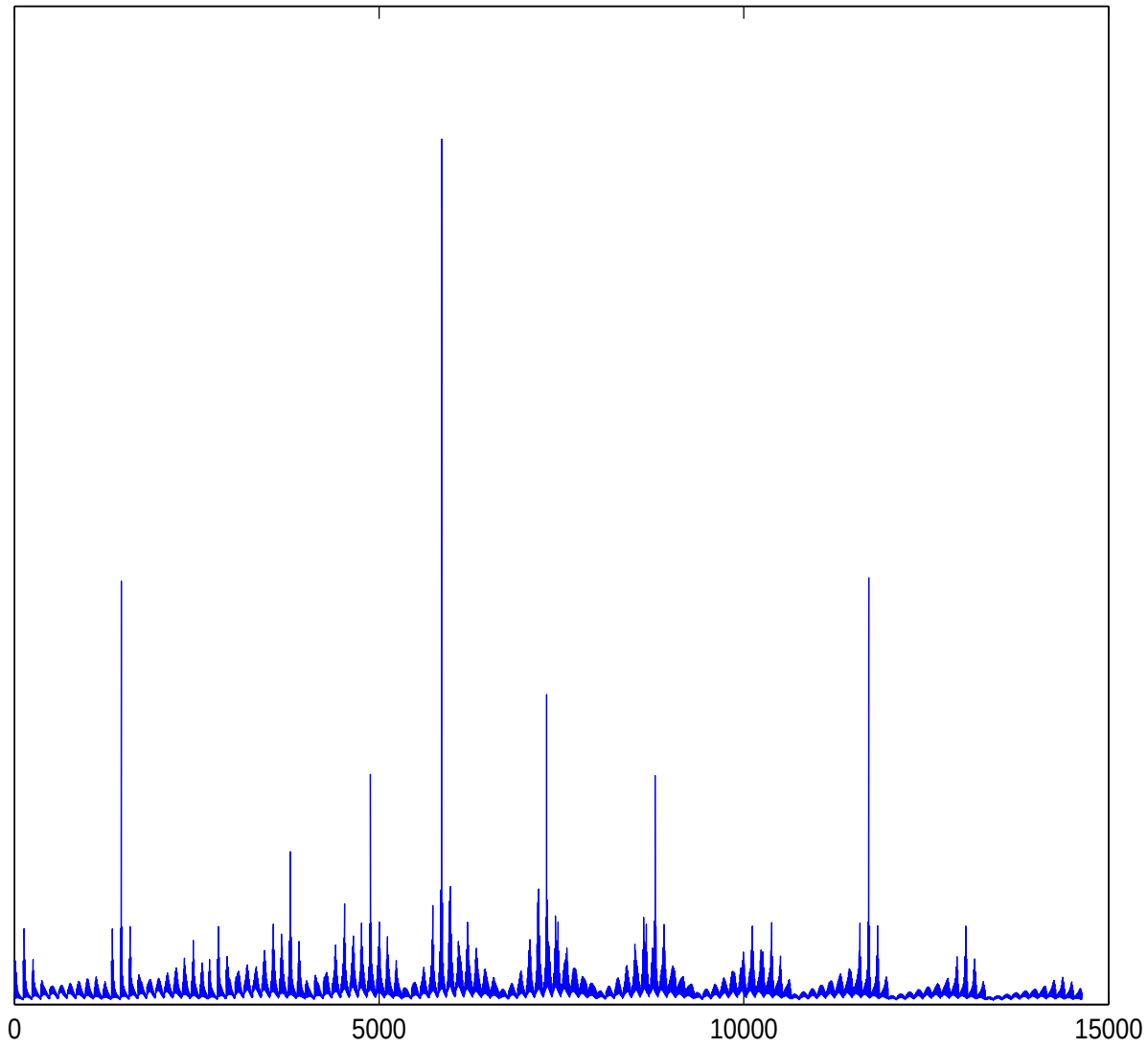
$$y(x_1,\ldots,x_4) = \sum_{i=1}^{m} \frac{1}{\sum_{j=1}^{4}(x_j - a_{ij})^2 + c_i}$$

The region of interest is $0 \leq x_j \leq 10$ and only integer values are considered.

Table 2 : Coefficients for Shekel's function ($m = 7$)

i	$a_{ij}, j = 1,\ldots,4$				c_i
1	4.0	4.0	4.0	4.0	0.1
2	1.0	1.0	1.0	1.0	0.2
3	8.0	8.0	8.0	8.0	0.2
4	6.0	6.0	6.0	6.0	0.4
5	3.0	7.0	3.0	7.0	0.4
6	2.0	9.0	2.0	9.0	0.6
7	5.0	5.0	3.0	3.0	0.3

Plot of Shekel 4 function



Performance of SELC : Shekel 4 function

- Four factors each at eleven levels (i.e. the 11 integers).
- Starting design is an orthogonal array - 4 columns from OA$(242, 11^{23})$.
- Forbidden arrays of order 3 are considered as order 1 or 2 becomes too restrictive.

Table 3 : % of success in identifying global maximum for different methods based on 1000 simulations

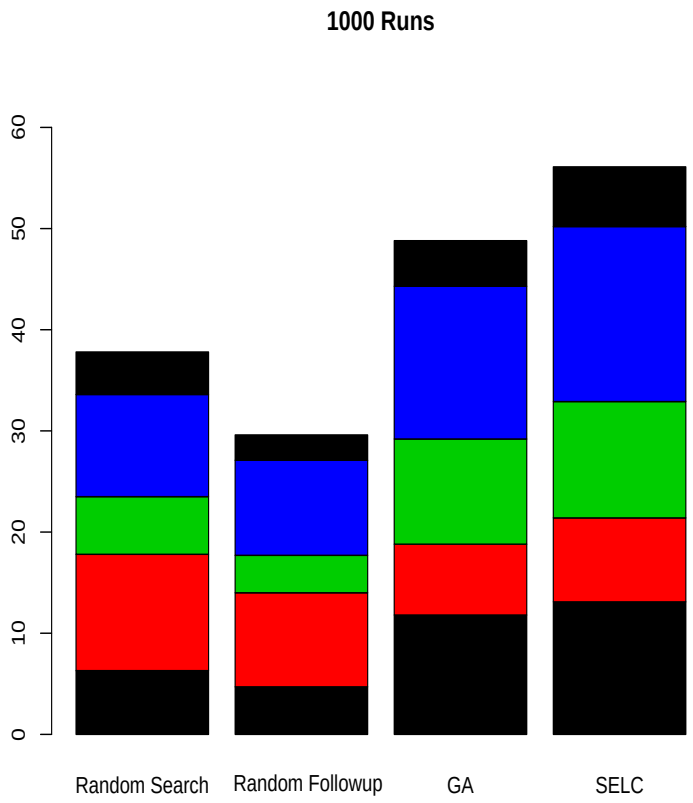
Run size = 1000

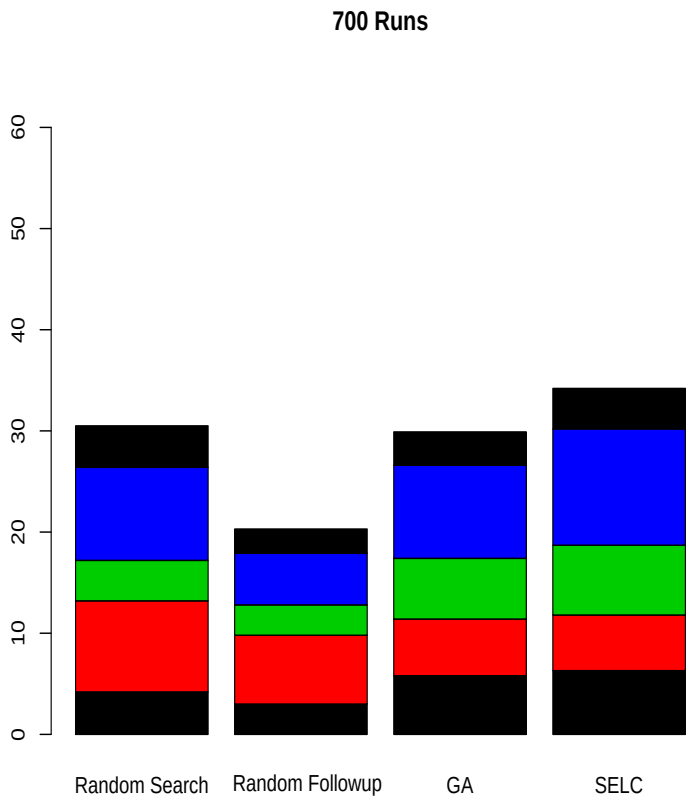
	Max	2nd best	3rd best	4th best	5th best	Total
Random Search	6.3	11.5	5.7	10.1	4.2	37.8
Random Followup	4.7	9.3	3.7	9.4	2.5	29.6
Genetic Algo	11.8	7.0	10.4	15.1	4.5	48.4
SELC	13.1	8.3	11.5	17.3	5.9	56.1

Run size = 700

	Max	2nd best	3rd best	4th best	5th best	Total
Random Search	4.2	9.0	4.0	9.2	4.1	30.5
Random Followup	3.0	6.8	3.0	5.1	2.4	20.3
Genetic Algo	5.8	5.6	6.0	9.2	3.3	29.9
SELC	6.3	5.5	6.9	11.5	4.0	34.2

Performance of SELC



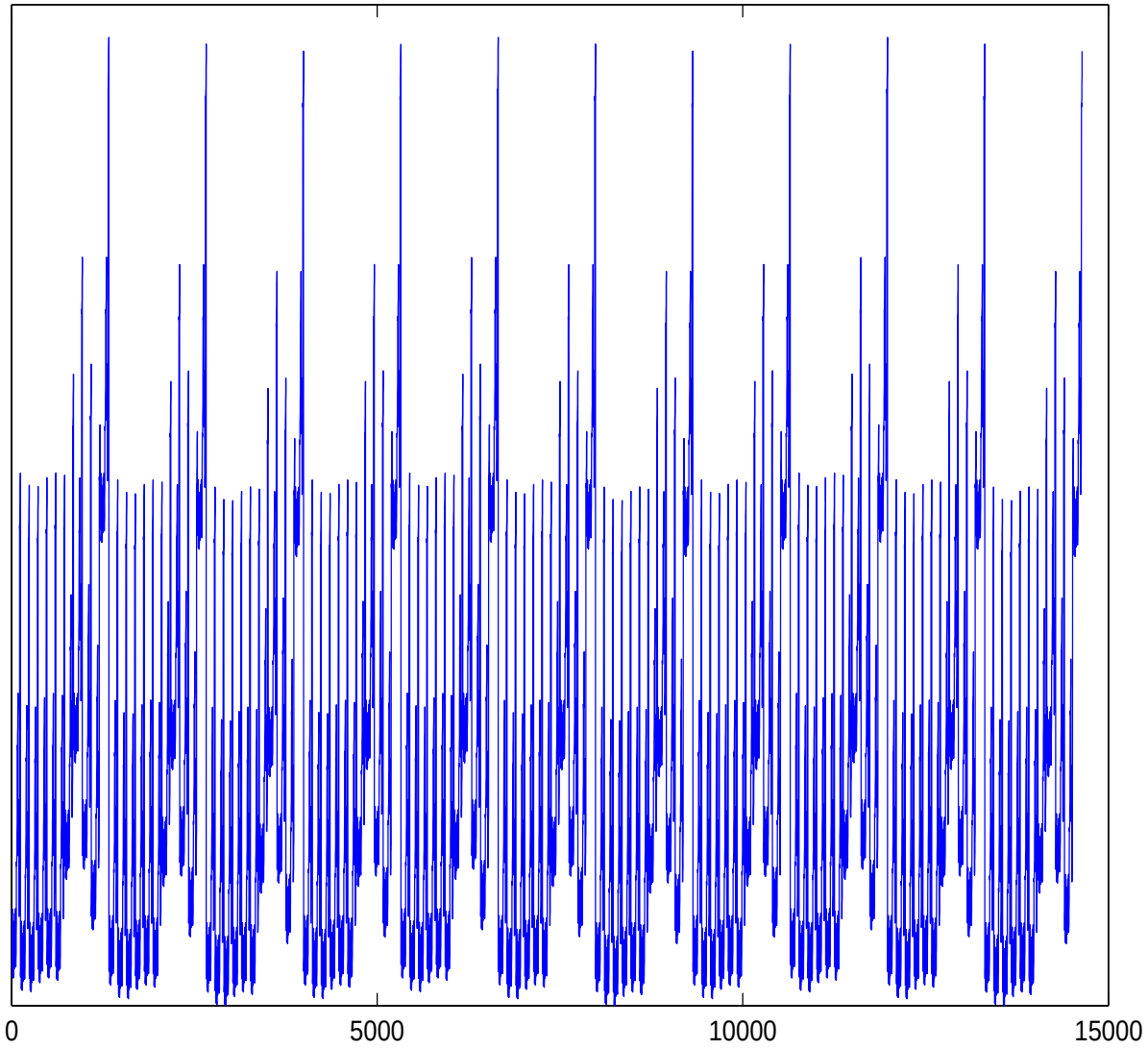


Example 2 (Levy and Montalvo)

$$y(x_1,\ldots,x_n) = \sin^2\left\{\pi\left(\frac{x_i+2}{4}\right)\right\} + \sum_{i=1}^{n-1}\left(\frac{x_i-2}{4}\right)^2\left\{1+10\sin^2\left(\pi\left(\frac{x_i+2}{4}\right)+1\right)\right.$$
$$\left. + \left(\frac{x_n-2}{4}\right)^2\left\{1+\sin^2\left(2\pi\left(x_n-1\right)\right)\right\},\right.$$

- Here $n = 4$.
- Only integer values of x_i's $(0 \leq x_i \leq 10)$ are considered.
- This again corresponds to an experiment with 4 factors each at 11 levels.

Plot of Levy's function

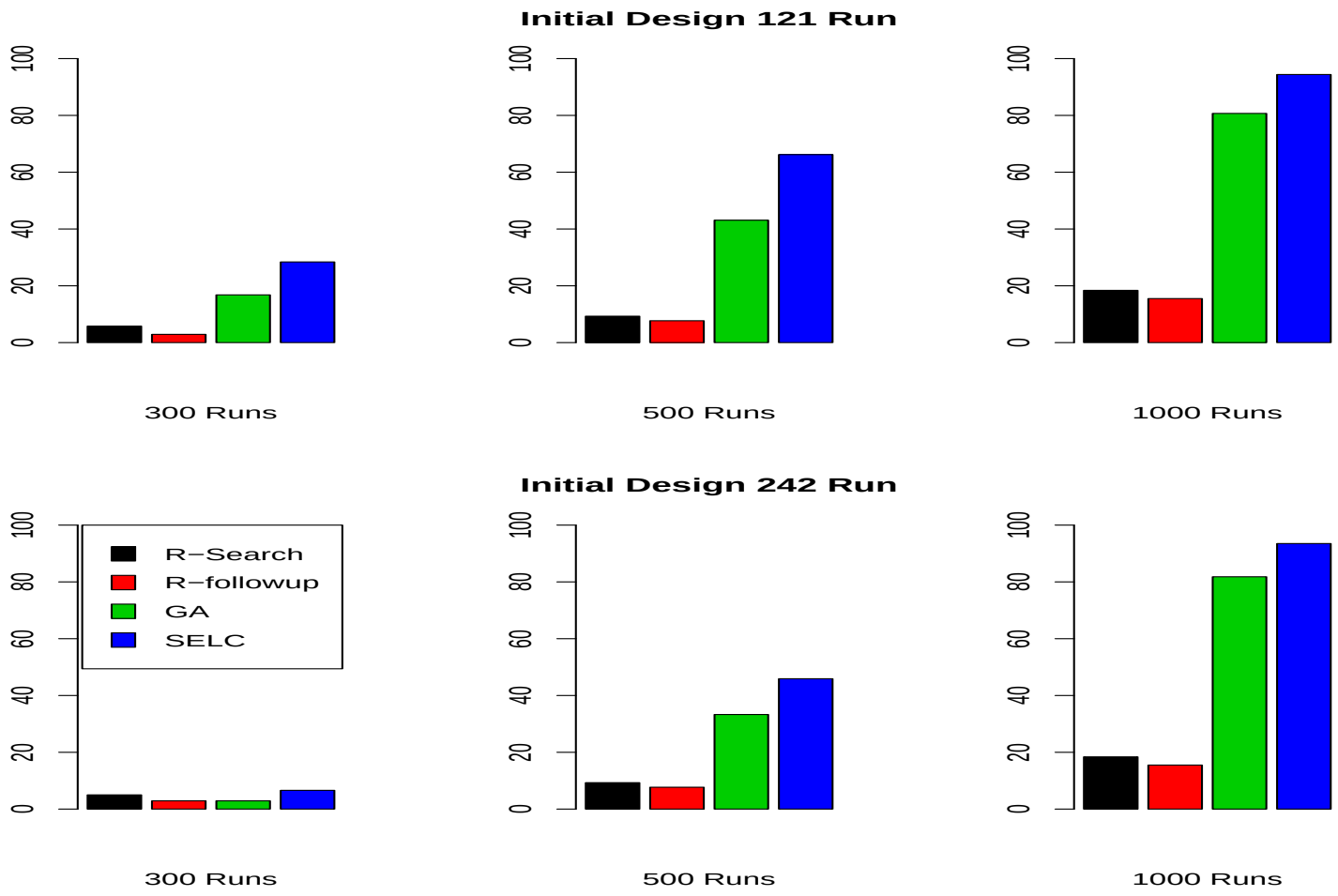


Performance of SELC

Table 4 : % of success in identifying global maximum for different methods based on 1000 simulations

	121-Run Design			242-Run Design		
Total Run Size	300	500	1000	300	500	1000
Random Search	5.8	9.3	18.4	5.0	9.3	18.4
Random Followup	2.9	7.7	15.5	2.9	7.7	15.5
Genetic Algo	16.8	43.1	80.7	2.9	33.3	81.8
SELC	28.4	66.2	94.4	6.6	45.9	93.5

Performance of SELC



Application

- SELC method was applied to a combinatorial chemistry problem where a combination of reagents was desired to maximize target efficacy (y).
- Target efficacy is measured by a compound's percent inhibition of activity for a specific biological screen.
- For this screen, a percent inhibition value of 50 or greater is an indicator of a promising compound. And, percent inhibition values of 95 or greater have a high probability of exhibiting activity in confirmation screening.
- Reagents can be added to 3 locations (A, B, and C) :

  $$2 \times 10 \times 14 = 280$$

  possible chemical entities.
- Due to resource limitations, only 25 compounds could be created.

Pharmaceutical Example (Cont.)

- **Forbidden Array:**
  - Forbidden array of order 2 was used.
  - Based on prior scientific knowledge, some combinations of reagents for this experiment were known to yield unfavorable percent inhibition values. These combinations of reagents were placed into the forbidden array prior to the experiment.
- **Starting Design:**
  - $2 \times 2 \times 3$ orthogonal array.
  - Want to have a small starting design. As resources allow to have only 25 runs, a 12 run starting design seems appropriate.
  - $2 \times 2 \times 3$ design is taken instead of $2 \times 3 \times 2$ design as there are more levels for C (as well as more "effective" levels).

Initial Design

- Next two Tables present the relative frequency of occurrence of the individual levels of factors B and C, respectively in the forbidden array.

Factor *B*

Level	1	2	3	4	5	6	7	8	9	10
Relative Freq. (in %)	3	3	26	4	29	5	10	1	5	14

Factor *C*

Level	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Rel. Freq.	8	7	7	4	5	4	4	3	8	5	16	11	8	8

Starting Experiment

#	A	B	C	y	
1	1	8	8	24	
2	1	9	8	-23	
3	2	8	8	34	
4	2	9	8	12	
5	1	8	3	63	*
6	1	9	3	21	
7	2	8	3	2	
8	2	9	3	9	
9	1	8	4	5	
10	1	9	4	-16	
11	2	8	4	49	*
12	2	9	4	5	

Weighted Mutation

- For B and C, not all levels are explored in the initial experiment. So if they turn out to be significant then its level is changed to any admissible level with some probability, and with higher probability to the promising levels.
- Negative values of y's are taken to be zero in calculating the mutation probabilities.
- In this case, B turns out to be significant after 13th run.

Weighted Mutation (Cont.)

- Let p_j be the probability with which the existing level is changed to level j.

$$p_8 = \frac{24+34+63+2+5+49+83+56+14+83}{1016} \times 0.75 + \frac{1}{10} \times 0.25$$

$$p_9 = \frac{0+12+21+9+0+5}{1016} \times 0.75 + \frac{1}{10} \times 0.25$$

$$p_j = \frac{1}{10} \times 0.25 \qquad \text{for all } j \neq 8,9$$

- Note the the sum of the positive values of y after first 13 runs is 1016.
- There are 10 levels of B which accounts for the $1/10$ in the above expression.
- The weights 0.75 and 0.25 are taken arbitrarily.

Follow-up Runs

The results from the subsequent runs are given below.

#	A	B	C	y	
13	2	8	10	83	*
14	2	3	4	65	*
15	2	1	4	107	*
16	2	2	10	49	
17	2	8	2	56	*
18	1	6	10	19	
19	2	2	4	60	*
20	2	10	10	39	
21	1	8	10	14	
22	2	6	8	90	*
23	2	6	10	64	*
24	2	1	1	-3	
25	2	2	5	63	*

Confirmatory Tests

- Clearly, the SELC method (with its slight modifications for this application) identified a rich set of compounds.
- In fact, all compounds run in the experiment were analyzed in a follow-up experiment where their IC_{50} values were determined. Compounds that were judged to be acceptable by the chemists are indicated with an asterisk.

Summary and Conclusions

- Good for relatively large space.
- Start with an **Orthogonal Design**. This helps identifying the important effects.
- **Bayesian variable selection** identifies the important factors.
- Follow-up runs are very flexible and **data-driven**.
- **Weighted Mutation uses sequential learning.**
- A novel blending of Design of Experiment ideas and Genetic Algorithms. SELC outperforms GA in many cases.
- Useful for many real-life examples.

Thank you