Computing efficient exact designs of experiments under cost constraints

Radoslav Harman

Faculty of Mathematics, Physics and Informatics
Comenius University in Bratislava, Slovakia

Design and Analysis of Experiments
Athens, GA
18 October 2012

Overview of the talk

- The model and the information matrix
- Exact and approximate designs of experiments (DoEs)
- Linear constraints on DoEs
  - Constraints on the size
  - Cost constraints
- Optimal linearly constrained DoEs
- Computing optimal exact cost-constrained DoEs
  - Exchange methods
  - Stochastic optimization methods
  - Rounding methods
  - Enumeration methods
- Conclusions

The model and the moment matrix

Design space $\mathfrak{X}$ is a finite set of labels of all permissible experimental conditions, under which the individual trials can be performed. Without loss of generality $\mathfrak{X} = \{1, ..., N\}$.

We will select a sequence $x_1, ..., x_n \in \mathfrak{X}$ of design points, perform the trials, and collect the observations $Y_1, ..., Y_n \in \mathbb{R}$ satisfying the **linear regression model**

$$Y_i = \beta^T f(x_i) + \epsilon_i, \;\; i = 1, ..., n,$$

- $\beta = (\beta_1, ..., \beta_m)^T \in \mathbb{R}^m$ ... unknown regression parameters
- $f = (f_1, ..., f_m)^T : \mathfrak{X} \to \mathbb{R}^m$ ... known regression functions
- $\epsilon_1, ..., \epsilon_n$ ... iid errors, $\mathrm{E}(\epsilon_i) = 0$, $\mathrm{Var}(\epsilon_i) = \sigma^2 \in (0, \infty)$.

Moment matrix (or "information matrix for β") is

$$\mathbf{M} = \sum_{i=1}^{n} f(x_i) f^T(x_i).$$

Exact and approximate designs

Exact design on $\mathfrak{X}$ is any $\xi \in \{0, 1, 2, ...\}^N$.

For $x \in \mathfrak{X}$ the value $\xi(x) \in \{0, 1, 2, ...\}$ represents the number of trials to be performed under the experimental conditions labeled by x. The set of all exact designs will be denoted by Ξ^E.

Approximate designs on $\mathfrak{X}$ is any $\xi \in [0, \infty)^N$.

Approximate (or "continuous") designs can be very useful as an auxiliary tool for computing exact designs. The set of all approximate designs will be denoted by Ξ. Clearly $\Xi^E \subset \Xi$.

The moment matrix of an experiment performed according to an exact or approximate design ξ:

$$\mathbf{M}(\xi) = \sum_{x \in \mathfrak{X}} \xi(x) f(x) f^T(x)$$

Exact and approximate designs

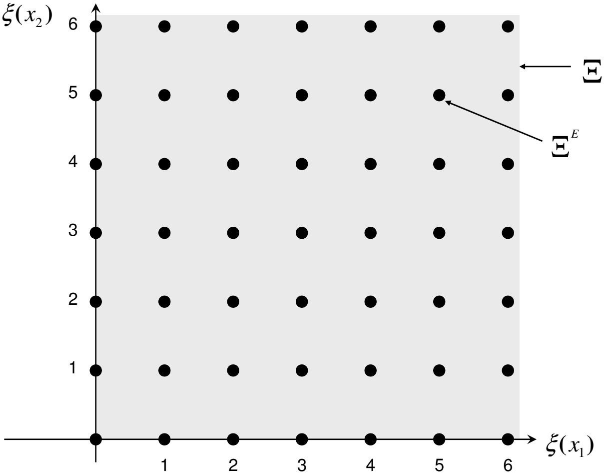

Figure: If $\mathfrak{X} = \{1, 2\}$ then Ξ^E is the set of all points in $\mathbb{R}^2$ with nonnegative integer coordinates, and Ξ is the first quadrant.

Linear constraints on designs

Let $\mathbf{C}$ be a given $K \times N$ matrix and $\mathbf{b}$ a given K-dimensional vector.

Linearly constrained design is any design ξ that satisfies K scalar inequalities

$$\mathbf{C}\xi \leq \mathbf{b}.$$

We will denote the set of all approximate linearly constrained designs by $\Xi_{\mathbf{C},\mathbf{b}}$ and the set of all exact linearly constrained designs by $\Xi^E_{\mathbf{C},\mathbf{b}}$.

If $K = 1$, $\mathbf{C} = \mathbf{1}_N = (1, ..., 1)$ and $\mathbf{b} = n \in \{1, 2, ...\}$ is a given "size" of the experiment, i.e., the limit on the number of trials, we obtain **size-constrained designs**

$$\sum_{x \in \mathfrak{X}} \xi(x) \leq n.$$

Most research in DoEs deals (only) with the problem of constructing exact or approximate size-constrained designs.

Special cases of general linear constraints

If $K = 1$, $\mathbf{C} = \mathbf{c} \in \mathbb{R}^N$ is a vector of "costs", and $\mathbf{b} = b \in (0, \infty)$ is a given limit on the total costs of the experiment, we obtain **cost-constrained designs**

$$\sum_{x \in \mathfrak{X}} c(x)\xi(x) \leq b.$$

The value $\mathbf{c}(x)$ can represent the costs of the material that is consumed or destroyed by the trial x, it can represent variable wages for the personnel performing the experiment, depending on x...

The general linear constraints can also represent

- **direct constraints**, i.e., upper and lower limits on the number of replications in the design points
- **marginal constraints**, i.e., upper limits on availability of material shared by a group of design points

See Cook and Fedorov, Statistics (1995).

Size-constrained designs

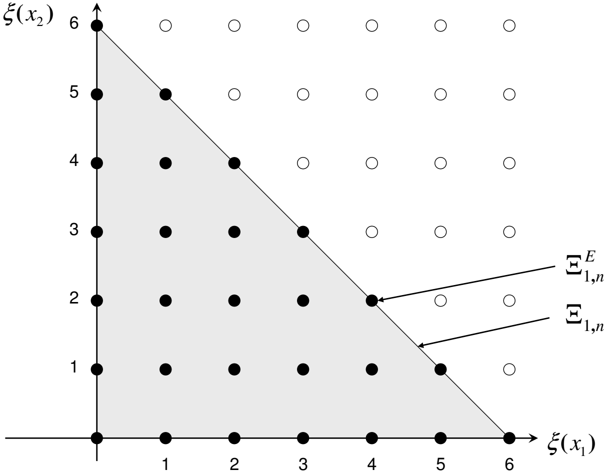

Figure: If $\mathfrak{X} = \{1, 2\}$ and $n = 6$ then $\Xi_{\mathbf{1},n}$ is the shaded triangle, and $\Xi^E_{\mathbf{1},n}$ is the set of all points with nonnegative integer coordinates in the shaded triangle.

Cost-constrained designs

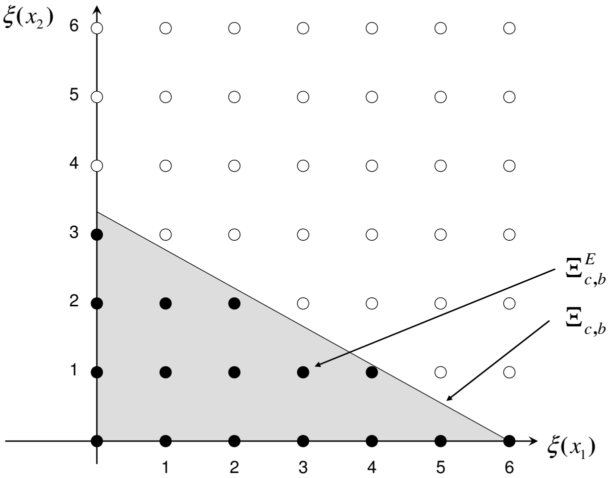

Figure: If $\mathfrak{X} = \{1, 2\}$, $\mathbf{c}(1) = 5$, $\mathbf{c}(2) = 9$, and $b = 30$, then $\Xi_{\mathbf{c},b}$ is the shaded triangle, and $\Xi^{E}_{\mathbf{c},b}$ is the set of points with nonnegative integer coordinates lying in the shaded triangle.

General linearly constrained designs

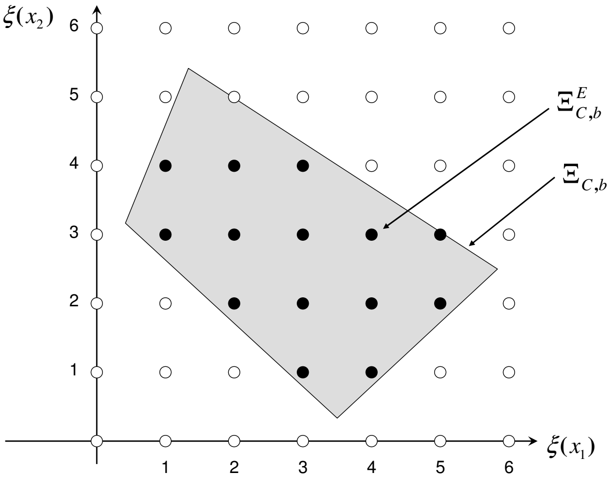

Figure: For $\mathfrak{X} = \{1, 2\}$, some 4×2 matrix $\mathbf{C}$ and some $\mathbf{b} \in \mathbb{R}^4$ the set $\Xi_{\mathbf{C},\mathbf{b}}$ is the shaded quadrilateral, and $\Xi^E_{\mathbf{C},\mathbf{b}}$ is the set of all points with nonnegative integer coordinates lying inside the shaded quadrilateral.

Criterion of optimality and optimal designs

Define an **optimality criterion**, i.e., a function

$$\Phi : \Xi \to [0, \infty)$$

measuring the "quality" of approximate and exact designs.

- **Φ-optimal approximate linearly constrained design:**

$$\xi_{\mathbf{C},\mathbf{b}} \in \mathrm{argmax}\{\Phi(\xi) : \xi \in \Xi_{\mathbf{C},\mathbf{b}}\}$$

- **Φ-optimal exact linearly constrained design:**

$$\xi^E_{\mathbf{C},\mathbf{b}} \in \mathrm{argmax}\{\Phi(\xi) : \xi \in \Xi^E_{\mathbf{C},\mathbf{b}}\}$$

For specific constraints we can define:

- Φ-optimal approximate size-n-constrained design $\xi_{\mathbf{1},n}$
- Φ-optimal exact size-n-constrained design $\xi^E_{\mathbf{1},n}$
- Φ-optimal approximate cost constrained design $\xi_{\mathbf{c},b}$
- Φ-optimal exact cost constrained design $\xi^E_{\mathbf{c},b}$

Optimal size-constrained designs

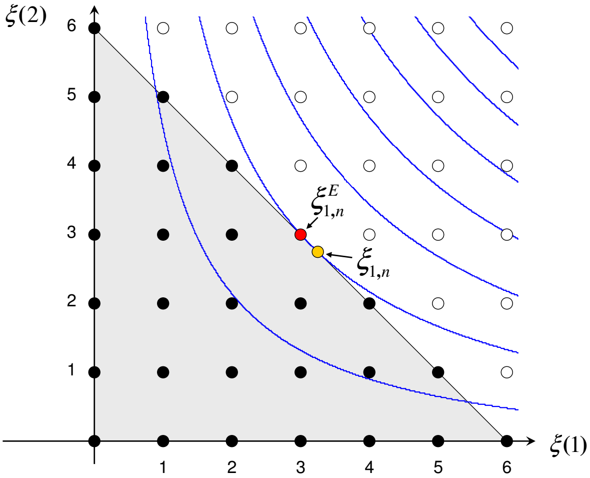

Figure: For $\mathfrak{X} = \{1, 2\}$, a hypothetical model and a criterion Φ, the red dot denotes the Φ-optimal exact size-constrained design and the orange dot denotes the Φ-optimal approximate size-constrained design.

Optimal cost-constrained designs

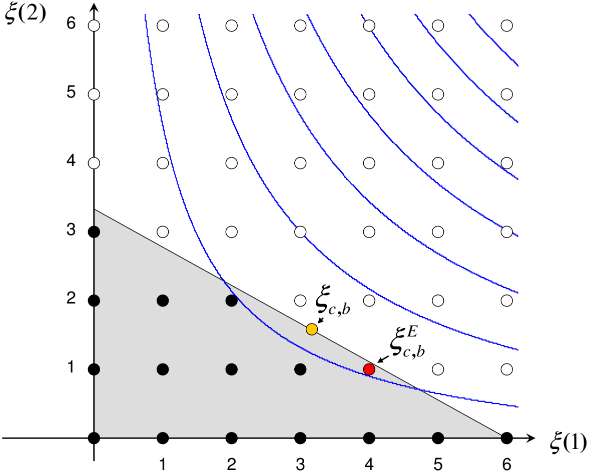

Figure: For $\mathfrak{X} = \{1, 2\}$, a hypothetical model and a criterion Φ, the red dot denotes the Φ-optimal exact cost-constrained design and the orange dot denotes the Φ-optimal approximate cost-constrained design.

Optimal general linearly constrained designs

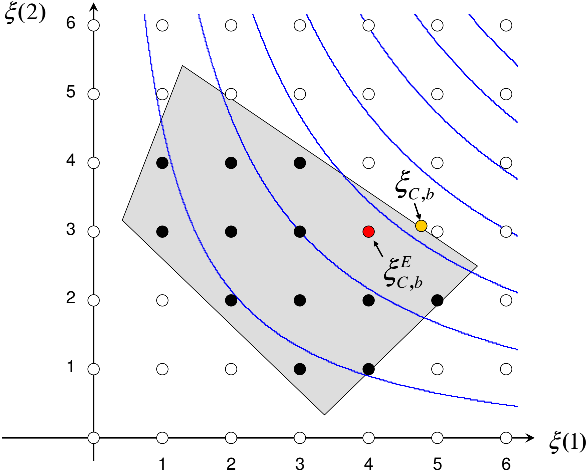

Figure: For $\mathfrak{X} = \{1, 2\}$, a hypothetical model, a criterion Φ, some $\mathbf{C}_{4\times 2}$, and some $\mathbf{b} \in \mathbb{R}^4$, the red dot is the Φ-optimal exact linearly constrained design and the orange dot is the Φ-optimal approximate linearly constrained design.

Computing exact linearly constrained designs

For a general **C**, for majority of models and criteria Φ the problem of computing a provably perfectly Φ-optimal exact cost-constrained design is NP-hard. (It comprises very hard combinatorial and combinatorial optimization problems as special cases.)

Most methods use heuristics that provide efficient but not always Φ-optimal designs.

Classes of methods for computing efficient cost constrained designs:

- Exchange (local search) methods
- Stochastic optimization methods
  - Simulated annealing
  - Genetic algorithms
- Rounding methods
- Enumeration methods
  - “Brute-force” complete enumeration methods
  - Partial enumeration methods, such as branch-and-bound

Exchange methods (general principle)

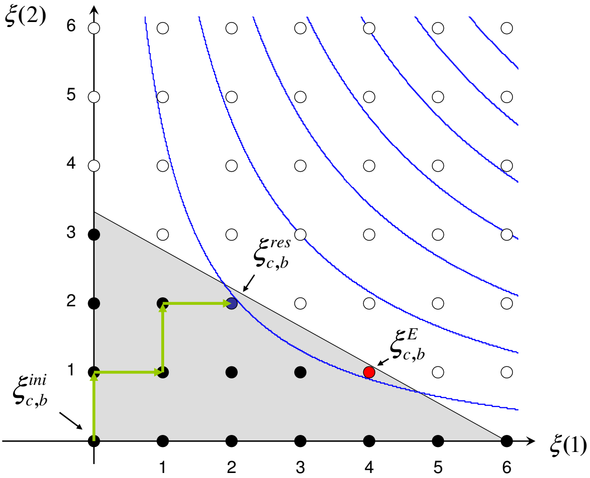

Figure: The design $\xi^{ini}_{\mathbf{c},b}$ is the initial design and $\xi^{res}_{\mathbf{c},b}$ is the resulting design. At each iteration, **a neighborhood** of the current design is searched and the best design (or the first better design) is chosen for the next iteration.

Exchange methods

Advantages:

- Can be fast even for very large models.
- Can be simple to implement.

Disadvantages:

- Sometimes stop in a locally optimal (i.e., not globally optimal) exact design.
- The principle is difficult to modify in a way that it produces efficient designs for some more general linear constraints.

A “barrier” method combined with an exchange algorithm for computing cost-constrained designs has been proposed in: Tack and Vandebroek, Journal of Statistical Planning and Inference (2004).

An exchange method for computing cost-constrained designs has been suggested in: Wright, Sigal, and Bailer, Journal of Agricultural, Biological, and Environmental Statistics (2010).

Stochastic optimization methods

Update a single permissible exact design (annealing-type methods) or a population of permissible exact designs (evolution-type methods) using a sequence of operations involving random “mutations”.

Advantages:

- Provide useful results even for very complex models.
- Can converge to a perfectly optimal exact design (in theory).

Disadvantages:

- Are usually slow compared to the exchange methods.
- Often do not provide perfectly optimal exact designs (in practice).

A genetic algorithm for cost-constrained design has been presented in: Park, Montgomery, Fowler and Borror, Quality and Reliability Engineering International (2006).

An annealing-type method for linearly constrained designs has been suggested in: Bachratá and Harman, Linstat Conference (2012).

Rounding methods (general principle)

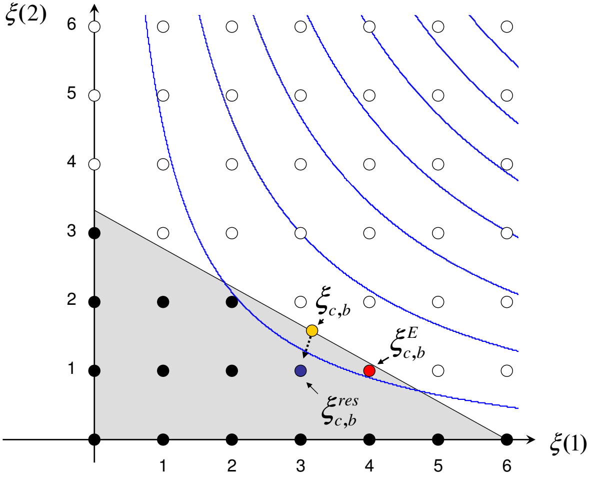

Figure: The optimal approximate design $\xi_{\mathbf{c},b}$ is “rounded” to a “closest” permissible exact design $\xi^{res}_{\mathbf{c},b}$.

Rounding methods

Advantages:

- Can be fast even for very large models.
- Can be simple to implement and use.

Disadvantages:

- Sometimes provide inefficient exact design, especially if the number of parameters is equal or only slightly higher than the number of trials.
- Can be difficult to adjust to some linear constraints.

A rounding method for cost-constrained designs has been suggested in: Sagnol, Discrete Applied Mathematics (2013).

A different method of using optimal approximate constrained designs to construct optimal exact designs has been suggested by Harman and Filová, submitted. (Can be used with general linear constraints.)

Branch-and-bound (general principle)

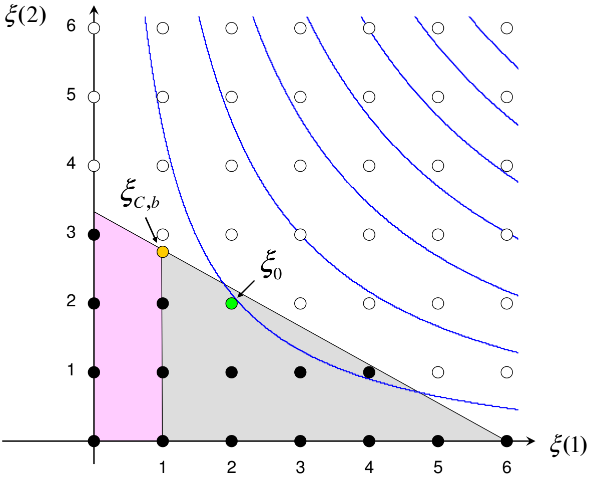

Figure: The green dot is a reference exact design and the orange dot is the Φ-optimal approximate design in the pink region. Since $\Phi(\xi_{\mathbf{C},\mathbf{b}}) < \Phi(\xi_0)$ we can skip enumerating the designs in the pink region.

Branch-and-bound methods

Advantages:

- Can be implemented to provide the complete list of all perfectly optimal exact designs.
- Can solve problems with general linear constraints on the design.

Disadvantages:

- Usually more difficult to implement.
- Very slow for large size problems.

The idea to use a branch-and-bound method for computing *D*-optimal exact designs of experiments: Welch, Technometrics (1982).

A branch-and-bound method for a problem with direct constraints on is used in: Ucinski and Patan, Journal of Global Optimization (2007).

Crucial requirement of application:

- Ability to compute the optimal value of the chosen optimality criterion on special constrained sets of *approximate* designs.

Criterion of $D_{\mathbf{A}}$-optimality

Let $\mathbf{A}$ be a full rank $m \times s$ matrix, $s \leq m$. $D_{\mathbf{A}}$-optimality:

$$\Phi_{\mathbf{A}}(\xi) = [\det(\mathbf{A}^T \mathbf{M}^-(\xi)\mathbf{A})]^{-1/s} \quad \text{if } \mathrm{span}(\mathbf{A}) \subseteq \mathrm{span}(\mathbf{M}(\xi)),$$

and $\Phi_{\mathbf{A}}(\xi) = 0$ otherwise, where $\mathbf{M}^-$ is a generalized inverse of $\mathbf{M}$.

- Under ξ, the value $\Phi_{\mathbf{A}}(\xi)$ is a measure of quality of estimation of $\mathbf{A}^T\beta$ by its best linear unbiased estimator (e.g., Pukelsheim (2006)).
- The most important special cases are the criteria of D-optimality if $\mathbf{A} = \mathbf{I}_m$, and linear optimality if $s = 1$.
- Using $D_{\mathbf{A}}$-optimality we can solve optimum design problems for the models with nuissance parameters, such as models with time or spatial trend, or block models.

It is possible to prove (Harman, submitted):

Theorem (Max-det programming characterization of $D_{\mathbf{A}}$-optimality)

$$\Phi_{\mathbf{A}}(\xi_{\mathbf{C},\mathbf{b}}) = \max\{[\det(\mathbf{N})]^{1/s} : \mathbf{N} \in \mathcal{S}^m_{++}, \mathbf{A}\mathbf{N}\mathbf{A}^T \preceq \mathbf{M}(\xi), \xi \in \Xi_{\mathbf{C},\mathbf{b}}\}$$

Can be computed by `sedumi` or `sdpt3`.

Example: Trend resistant cost constrained designs

$$Y_i = \tau_{t(i)} + \theta_1 p_0(i) + \theta_2 p_1(i) + \theta_3 p_2(i) + \theta_4 p_3(i) + \epsilon_i,\ i = 1, ..., n$$

- $t(i) \in \{1, 2, 3\}$ ... treatment applied to the time point i
- τ_1, τ_2, τ_3 ... effects of treatments
- $\theta_1, \theta_2, \theta_3, \theta_4$ ... (nuisance) parameters of the time trend
- p_0, p_1, p_2, p_3 ... polynomials of degrees $0, 1, 2, 3$
- ϵ_i ... iid random errors

The aim is to select the D-optimal sequence of treatments that minimizes the determinant of the variance-covariance matrix of the BLUE of contrasts $\tau_2 - \tau_1$ and $\tau_3 - \tau_1$. (Closely related problems have been solved in Atkinson and Donev, Technometrics (1996).)

It turns out that this can be equivalently formulated as a problem of constructing a $D_{\mathbf{A}}$-optimal exact marginally constrained design for some matrix $\mathbf{A}$, some design space $\mathfrak{X}$ with $3n$ design points, specific vector f of regressors, and a vector $\beta \in \mathbb{R}^7$ of unknown parameters.

Example: Trend resistant designs

The list of all D-optimal design sequences of treatments for estimating $\tau_2 - \tau_1$ and $\tau_3 - \tau_1$ can be computed using a branch-and-bound method with the max-det characterization of $D_{\mathbf{A}}$-optimality.

For instance, for $n = 18$ time points the set of D-optimal designs consists of exactly 12 design, which can be obtained by any relabeling of the treatments of the sequences:

		3	3		3							3		3	3		
	2			2			2			2			2			2	
1						1		1	1		1						1

		3	3			3					3			3	3		
	2			2			2			2			2			2	
1					1			1	1			1					1

Example: Trend resistant cost constrained designs

Assume that we want to construct a *D*-optimal cost-constrained sequence of treatments for estimating $\tau_2 - \tau_1$ and $\tau_3 - \tau_1$. It is simple to add the cost constraints to the branch-and-bound algorithm.

For instance, if $n = 18$, the costs of treatments $1, 2, 3$ are $0, 1, 2$ price units, and the budget limit is 15, there are exactly two *D*-optimal treatment sequences:

		3						3							3		
2						2	2			2	2						2
	1		1	1	1				1			1	1	1		1	

		3							3						3		
2						2	2			2	2						2
	1		1	1	1			1				1	1	1		1	

Conclusions

- Linearly constrained designs (especially cost, marginal, and directly constrained designs) can be found in many applications.
- Often, it is non-trivial to adjust the standard methods of computing size-constrained designs to the problem with more general linear constraints. The adjustments usually use specifics of some particular type of linear constraints.
- For the construction of efficient but not always optimal linearly constrained designs in large size models we can use methods such as modified exchange algorithms, stochastic optimization methods, and rounding methods.
- For small and medium size problems it is possible to use "intelligent" enumeration methods that can provide a complete list of perfectly optimal exact designs under general linear constraints. As a fundamental element they use methods for constructing optimal approximate linearly constrained design.

Thank you!