DEPARTMENT OF ENGINEERING MANAGEMENT

An algorithmic framework for generating optimal two-stratum experimental designs

Daniel Palhazi Cuervo, Peter Goos & Kenneth Sörensen

UNIVERSITY OF ANTWERP
Faculty of Applied Economics

City Campus
Prinsstraat 13, B.226
B-2000 Antwerp
Tel. +32 (0)3 265 40 32
Fax +32 (0)3 265 47 99
www.uantwerpen.be

FACULTY OF APPLIED ECONOMICS

DEPARTMENT OF ENGINEERING MANAGEMENT

An algorithmic framework for generating optimal two-stratum experimental designs

Daniel Palhazi Cuervo, Peter Goos & Kenneth Sörensen

RESEARCH PAPER 2016-003
FEBRUARY 2016

University of Antwerp, City Campus, Prinsstraat 13, B-2000 Antwerp, Belgium
Research Administration – room B.226
phone: (32) 3 265 40 32
fax: (32) 3 265 47 99
e-mail: joeri.nys@uantwerpen.be

D/2016/1169/003

An algorithmic framework for generating optimal two-stratum experimental designs

Daniel Palhazi Cuervo[1], Peter Goos[1,2]and Kenneth Sörensen[1]

[1]University of Antwerp, Faculty of Applied Economics, Department of Engineering Management
[2]KU Leuven, Faculty of Bioscience Engineering, Department of Biosystems & LSTAT

February 23, 2016

Abstract

Two-stratum experiments are widely used in case a complete randomization is not possible. In some experimental scenarios, there are constraints that limit the number of observations that can be made under homogeneous conditions. In other scenarios, there are factors whose levels are hard or expensive to change. In both of these scenarios, it is necessary to arrange the observations in different groups. Moreover, it is important that the analysis performed accounts for the variation in the response variable due to the differences between the groups. The most common strategy for the design of these kinds of experiments is to consider groups of equal size. The number of groups and the number of observations per group are usually defined by the constraints that limit the experimental scenario. We argue, however, that these constraints do not define the design itself, but should be considered only as upper bounds. The number of groups and the number of observations per group should be chosen not only to satisfy the experimental constraints, but also to maximize the quality of the experiment. In this paper, we propose an algorithmic framework to generate optimal designs for two-stratum experiments in which the number of groups and the number of observations per group are limited only by upper bounds. The results of an extensive set of computational simulations show that this additional flexibility in the design generation process can significantly improve the quality of the experiments. Moreover, the results show that the grouping configuration of an optimal design depends on the characteristics of the two-stratum experiment, namely, the type of experiment, the model to be estimated and the optimality criterion considered. This is certainly a strong argument in favour of using algorithmic techniques that are able to identify not only the best factor-level configuration for each experimental run, but also the best grouping configuration.

Keywords: Blocked experiments, Split-plot experiments, Two-stratum experiments, $\mathcal{D}$-optimality criterion, $\mathcal{D}_s$-optimality criterion, $\mathcal{I}$-optimality criterion, $\mathcal{I}_d$-optimality criterion, Coordinate-exchange algorithm, Variable neighborhood search.

1 Introduction

Randomization is one of the most important principles in experimental design. In order to carry out a traditional experiment, it is paramount that the experimental observations are made independently, in a random order. In case the experiment involves factors defining the settings of an apparatus, it is necessary that these settings are reset every time a new experimental observation is to be made. The reason for this is that randomization offers protection against systematic bias due to extraneous factors that are not under control of the experimenter. There exist experimental scenarios, however, in which a complete randomization is not

possible.

One scenario involves cases where not all observations can be made under homogeneous conditions; for example, when only a limited number of observations can be made per day, or when different batches of raw material are used to study an industrial process. Trinca and Gilmour (2000a,b) describe an experiment regarding the baking of pastry dough with this limitation. Researchers in the food industry were interested in studying the influence of three factors on the quality of the pastry produced in a pilot plant. These factors were the initial moisture content of the dough, the screw speed of the mixer and the feed flow rate of water being added to the mix. The researchers performed an experiment with 28 observations. However, only 4 observations could be performed per day and an important day-to-day variation was expected. There is substantial literature on experimental scenarios with similar characteristics (see, for example, Chasalow (1992) and Khuri (1992)). For these cases, suitable experimental designs require to group the observations made under similar experimental conditions. Moreover, a proper analysis requires to model the variation of the response due to the differences between the groups. These types of experiments are called *blocked experiments*.

Another scenario involves cases with factors that are hard, time-consuming or expensive to change. Anderson and McLean (1974) discuss an example regarding the production of steel alloys using an industrial furnace. Among other factors, the experiment studied the influence of the furnace temperature on the strength of the steel alloys. Increasing or decreasing the temperature of the furnace was very cumbersome. Therefore, each temperature change considerably increased the duration of the experiment. Other experimental scenarios involving factors with similar characteristics are described by Letsinger et al. (1996), Trinca and Gilmour (2001), Jones and Goos (2007) and Gilmour and Goos (2009). In these situations, it is common that the person in charge of the experiment establishes a practical order of observations that avoids these inconvenient factor-level changes. Since the factor levels are not reset every time, this results in groups of dependant or correlated observations. A better approach is to explicitly consider this correlation structure in the experiment, and generate a design that groups the observations in which the levels of the factors that are inconvenient to change remain constant. The experiments that address scenarios with these characteristics are called *split-plot experiments*. There are, additionally, other types of experiments that involve more complex configurations of hard-to-change factors; for example, *split-split-plot experiments* (Jones and Goos, 2009), *strip-plot experiments* (Arnouts et al., 2010; Miller, 1997) and *staggered-level experiments* (Arnouts and Goos, 2012, 2015).

Blocked and split-plot experiments are referred to as *two-stratum experiments* in reference to the hierarchical structure of levels or strata they involve: every unit in the upper stratum is composed of one or more units in the lower stratum. The upper stratum comprises the groups of observations and the lower stratum comprises the individual observations. Observe that we refer to *groups* of observations as a generic term that can be applied to any two-stratum experiment. There is, however, specific terminology for each type of experiment. In the literature regarding blocked experiments, groups are referred to as *blocks*, while, in the split-plot literature, they are referred to as *whole plots*. In this paper, we use the appropriate terminology when discussing a specific type of experiment. We use the term *group* for statements that are valid for any two-stratum experiment.

The use of two-stratum experiments is mainly due to logistic or experimental constraints. However, there are also strong statistical reasons to consider an experiment with restricted randomization. Anbari and Lucas (1994, 2008) show how conducting factorial experiments in a split-plot fashion results in lower maximum prediction variances than when carried out in a completely randomized way. Goos and Vandebroek (2001a, 2004) also show that split-plot designs can outperform completely randomized designs in terms of statistical efficiency, and recommend their use even if no-hard-to-change factors are present. Two-stratum experiments not only are easier and more cost-effective to carry out, but they are also very efficient from a statistical viewpoint in comparison to completely randomized experiments. For these reasons, they have become in-

creasingly popular in the last few years and the scientific community is strongly advocating their use (Jones and Nachtsheim, 2009).

The most common strategy for the design of two-stratum experiments is to consider homogeneous grouping configurations, i.e., to consider groups with equal numbers of observations. The number of groups and the number of observations per group are typically determined by logistical constraints. A straightforward approach is therefore to use as many groups and/or as many observations per group as the constraints allow. However, these constraints do not define the experimental scenario itself, but only impose upper bounds on it. In many cases, there exist several alternative designs with different grouping configurations that satisfy the constraints. Nevertheless, experimenters rarely evaluate these alternatives and usually stick to the most straightforward design. We believe that an important characteristic of a good experimental design is that it not only has a grouping configuration that complies with the constraints, but that it also has optimal statistical properties. We therefore consider very important to search for the best grouping configuration.

Considering an experimental scenario to be limited only by upper bounds makes the task of generating an optimal design much more demanding. Besides determining the factor settings for each observation and the group each observation is assigned to, it is also necessary to determine the number of groups and the size of each group. This additional flexibility in the design generation process can significantly improve the quality of the experiment. As shown by Goos (2006) and Atkinson et al. (2007), several experimental scenarios exist in which the optimal design is composed of heterogeneous groups with different numbers of observations. The characteristics of the optimal design, however, strongly depend on the experimental scenario. For this reason, it is necessary to use algorithmic techniques that are capable of identifying both the best grouping configuration and the best factor-level combinations for each experimental run.

In this paper, we propose an algorithmic framework for the optimal design of two-stratum experiments where the number of groups and number of observations per group are limited only by upper bounds. The best grouping configuration (namely, the number of groups and the size of each group) is automatically identified by the framework. One main advantage of this approach is that it does not make use of a set of candidate points, as it is fully based on the coordinate-exchange algorithm. For this reason, it is able to generate designs for experiments with larger numbers of factors and observations in shorter execution times than point-exchange algorithms. Additionally, we carry out an extensive computational simulation with two main purposes. First, we want to study the dependence of the best grouping configuration of an optimal design on the characteristics of the experiment, the model to be estimated and the optimality criterion considered. Second, we want to analyze the impact of the experimental constraints on the statistical efficiency of the optimal designs generated. To this end, we do not only consider the traditional $\mathcal{D}$-, $\mathcal{D}_s$- and $\mathcal{I}$-optimality criteria, but also the more recently introduced $\mathcal{I}_d$-optimality criterion. To our knowledge, this is the first time that $\mathcal{D}_s$- and $\mathcal{I}_d$-optimal designs for two-stratum experiments with different grouping configurations are studied. In order to show the results of the computational simulations, we make use of different heat maps. This kind of graphical representation allows for a quick and easy-to-understand analysis.

This paper is organized as follows. Section 2 reviews the statistical model used to analyze data from two-stratum experiments and the optimality criteria considered to generate optimal designs. Section 3 discusses a small motivating example that shows the strong impact of the grouping configuration on the quality of an experimental design. Section 4 provides a brief overview of both analytical and algorithmic techniques to generate optimal designs for two-stratum experiments. The proposed algorithmic framework is described in Section 5 and the computational simulations are discussed in Section 6. Section 7 compares the designs generated by our framework to those generated by other algorithmic techniques. Section 8 compares the statistical efficiency of optimal designs for two-stratum experiments to that of optimal designs for completely randomized experiments. The final conclusions and some recommendations for future research are presented in Section 9.

2 Statistical model and design optimality criteria

2.1 Model

In general, the statistical model used for analysing data from two-stratum experiments involves random effects for the different groups. Since the observations carried out in the same group are statistically dependent, it is necessary to consider a compound symmetric error structure in order to properly analyze the data. Assume that a two-stratum experiment consists of N experimental observations. These observations are grouped in K groups of sizes $N_1, \ldots, N_K$ such that $\sum_{k=1}^{K} N_k = N$. Additionally, assume that the experiment involves H hard-to-change factors and E easy-to-change factors. Within the k-th group, the i-th observation Y_{ki} is written as

$$Y_{ki} = \boldsymbol{f}'(\boldsymbol{w}_k, \boldsymbol{s}_{ki})\boldsymbol{\beta} + \delta_k + \varepsilon_{ki}, \tag{1}$$

where $\boldsymbol{w}_k$ is the $H \times 1$ vector with the settings of the hard-to-change factors for all the observations in the k-th group, $\boldsymbol{s}_{ki}$ is the $E \times 1$ vector with the settings of the easy-to-change factors for the i-th observation in the k-th group, $\boldsymbol{f}(\boldsymbol{w}_k, \boldsymbol{s}_{ki})$ represents the $P \times 1$ vector containing the polynomial expansion of the factor levels, $\boldsymbol{\beta}$ is the $P \times 1$ vector containing the P model parameters, δ_k represents the k-th random group effect and ε_{ki} is the residual error corresponding to the i-th observation in the k-th group. It is assumed that the group effects and the residual errors are independent and normally distributed with zero mean, that the group effects have variance σ_δ^2 and that the residual errors have variance σ_ε^2. The dependence between observations $Y_{k1}, \ldots, Y_{kN_k}$ in the k-th group is described by the $N_k \times N_k$ variance-covariance matrix

$$\boldsymbol{V}_k = \sigma_\varepsilon^2 \begin{bmatrix} 1 + \sigma_\delta^2/\sigma_\varepsilon^2 & \sigma_\delta^2/\sigma_\varepsilon^2 & \ldots & \sigma_\delta^2/\sigma_\varepsilon^2 \\ \sigma_\delta^2/\sigma_\varepsilon^2 & 1 + \sigma_\delta^2/\sigma_\varepsilon^2 & \ldots & \sigma_\delta^2/\sigma_\varepsilon^2 \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_\delta^2/\sigma_\varepsilon^2 & \sigma_\delta^2/\sigma_\varepsilon^2 & \ldots & 1 + \sigma_\delta^2/\sigma_\varepsilon^2 \end{bmatrix}, \tag{2}$$

where the variance ratio $\sigma_\delta^2/\sigma_\varepsilon^2$ measures the extent to which observations in the same group are correlated. In matrix notation, the model can be expressed as

$$\boldsymbol{Y} = \boldsymbol{f}(\boldsymbol{X})\boldsymbol{\beta} + \boldsymbol{B}\boldsymbol{\delta} + \boldsymbol{\varepsilon}, \tag{3}$$

where $\boldsymbol{Y}$ is the $N \times 1$ vector containing the responses, $\boldsymbol{X}$ is the $N \times (H + E)$ design matrix containing the factor levels at each observation, $\boldsymbol{f}(\boldsymbol{X})$ is the $N \times P$ extended design matrix composed of the polynomial expansions for each observation, $\boldsymbol{B}$ is the $N \times K$ matrix in which the element b_{ik} is one if the i-th observation belongs to the k-th group and zero otherwise, $\boldsymbol{\delta}$ is the $K \times 1$ vector of group effects and $\boldsymbol{\varepsilon}$ is the $N \times 1$ vector containing the residual error of each observation. If we assume that the observations in $\boldsymbol{Y}$ are arranged per group, such that

$$\boldsymbol{Y} = \begin{bmatrix} Y_{11} & \ldots & Y_{1N_1} & Y_{21} & \ldots & Y_{2N_2} & \ldots & Y_{K1} & \ldots & Y_{KN_K} \end{bmatrix}', \tag{4}$$

then the matrices $\boldsymbol{f}(\boldsymbol{X})$ and $\boldsymbol{B}$ are of the form

$$\boldsymbol{f}(\boldsymbol{X}) = \begin{bmatrix} \boldsymbol{f}(\boldsymbol{X}_1) \\ \boldsymbol{f}(\boldsymbol{X}_2) \\ \vdots \\ \boldsymbol{f}(\boldsymbol{X}_K) \end{bmatrix} = \begin{bmatrix} \boldsymbol{f}'(\boldsymbol{w}_1, \boldsymbol{s}_{11}) \\ \vdots \\ \boldsymbol{f}'(\boldsymbol{w}_1, \boldsymbol{s}_{1N_1}) \\ \boldsymbol{f}'(\boldsymbol{w}_2, \boldsymbol{s}_{21}) \\ \vdots \\ \boldsymbol{f}'(\boldsymbol{w}_2, \boldsymbol{s}_{2N_2}) \\ \vdots \\ \boldsymbol{f}'(\boldsymbol{w}_K, \boldsymbol{s}_{K1}) \\ \vdots \\ \boldsymbol{f}'(\boldsymbol{w}_K, \boldsymbol{s}_{KN_K}) \end{bmatrix} \text{ and } \boldsymbol{B} = \begin{bmatrix} 1 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 \end{bmatrix},$$

where $\boldsymbol{X}_k$ is the subdesign matrix corresponding to the k-th group. For this order of observations, the variance-covariance matrix of the vector $\boldsymbol{Y}$ has the form

$$\boldsymbol{V} = \begin{bmatrix} \boldsymbol{V}_1 & \boldsymbol{0}_{N_1 \times N_2} & \dots & \boldsymbol{0}_{N_1 \times N_K} \\ \boldsymbol{0}_{N_2 \times N_1} & \boldsymbol{V}_2 & \dots & \boldsymbol{0}_{N_2 \times N_K} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{0}_{N_K \times N_1} & \boldsymbol{0}_{N_K \times N_2} & \dots & \boldsymbol{V}_K \end{bmatrix}, \tag{5}$$

where $\boldsymbol{0}_{i \times j}$ is a zero matrix with i rows and j columns. The maximum likelihood estimator of the unknown parameter values $\boldsymbol{\beta}$ is the generalized least squares (GLS) estimator

$$\hat{\boldsymbol{\beta}} = (\boldsymbol{f}'(\boldsymbol{X})\boldsymbol{V}^{-1}\boldsymbol{f}(\boldsymbol{X}))^{-1}\boldsymbol{f}'(\boldsymbol{X})\boldsymbol{V}^{-1}\boldsymbol{Y}, \tag{6}$$

with variance-covariance matrix

$$\text{var}(\hat{\boldsymbol{\beta}}) = (\boldsymbol{f}'(\boldsymbol{X})\boldsymbol{V}^{-1}\boldsymbol{f}(\boldsymbol{X}))^{-1}. \tag{7}$$

2.2 Design optimality criteria

The goal of an optimal experimental design is to maximize the amount of information the experiment generates. This information is quantified using an optimality criterion. In this section, we describe the optimality criteria we consider for the optimal design of two-stratum experiments. A key feature of all criteria is that they depend on the variance ration $\sigma^2_\delta/\sigma^2_\varepsilon$, through the variance-covariance matrix Y. For the sake of simplicity, we assume this ratio to be equal to 1 in the remaining sections of the paper.

2.2.1 $\mathcal{D}$-optimality criterion

The $\mathcal{D}$-optimality criterion is the most commonly used criterion in the literature. It seeks to obtain precise parameter estimates by minimizing the determinant of the variance-covariance matrix of the GLS estimator $\hat{\boldsymbol{\beta}}$. This is equivalent to maximizing the determinant of the information matrix

$$\mathcal{D}(\boldsymbol{X}) = |\boldsymbol{f}'(\boldsymbol{X})\boldsymbol{V}^{-1}\boldsymbol{f}(\boldsymbol{X})|. \tag{8}$$

A design that maximizes this determinant value is called a $\mathcal{D}$-optimal design. In order to compare the quality of two designs with design matrices $\boldsymbol{X}_A$ and $\boldsymbol{X}_B$, we calculate the $\mathcal{D}$-efficiency of the first design relative to the second using the expression

$$\mathcal{D}\text{-eff}(\boldsymbol{X}_A, \boldsymbol{X}_B) = 100 \left(\frac{\mathcal{D}(\boldsymbol{X}_A)}{\mathcal{D}(\boldsymbol{X}_B)}\right)^{1/P} = 100 \left(\frac{|\boldsymbol{f}'(\boldsymbol{X}_A)\boldsymbol{V}_A^{-1}\boldsymbol{f}(\boldsymbol{X}_A)|}{|\boldsymbol{f}'(\boldsymbol{X}_B)\boldsymbol{V}_B^{-1}\boldsymbol{f}(\boldsymbol{X}_B)|}\right)^{1/P}, \tag{9}$$

where $\boldsymbol{V}_A$ is the variance-covariance matrix of the observations produced by the design with design matrix $\boldsymbol{X}_A$ and $\boldsymbol{V}_B$ is the variance-covariance matrix of the observations produced by the design with design matrix $\boldsymbol{X}_B$.

2.2.2 $\mathcal{D}_s$-optimality criterion

The $\mathcal{D}_s$-optimality criterion also seeks to obtain precise parameter estimates; however, it only focuses on a subset of parameters. In this paper, we consider this subset, $\boldsymbol{\beta}^\dagger$, to include all parameters except the intercept of the model. As a result, our $\mathcal{D}_s$-optimality criterion minimizes the determinant of the variance-covariance matrix of the estimator $\hat{\boldsymbol{\beta}}^\dagger$,

$$\mathcal{D}_s(\boldsymbol{X}) = |\text{var}(\hat{\boldsymbol{\beta}})_{2,2}|^{1/(P-1)} = |(\boldsymbol{f}'(\boldsymbol{X})\boldsymbol{V}^{-1}\boldsymbol{f}(\boldsymbol{X}))^{-1}_{2,2}|^{1/(P-1)}, \tag{10}$$

where $\text{var}(\hat{\boldsymbol{\beta}})_{2,2}$ is the matrix that results from eliminating the first row and the first column of the variance-covariance matrix of the overall estimator $\hat{\boldsymbol{\beta}}$. Note that the row and the column eliminated correspond to the intercept, which is not included in $\boldsymbol{\beta}^\dagger$. We calculate the $\mathcal{D}_s$-efficiency of a design with design matrix $\boldsymbol{X}_A$ relative to another design with design matrix $\boldsymbol{X}_B$ as

$$\mathcal{D}_s\text{-eff}(\boldsymbol{X}_A, \boldsymbol{X}_B) = 100\,\frac{\mathcal{D}_s(\boldsymbol{X}_B)}{\mathcal{D}_s(\boldsymbol{X}_A)} = 100\,\frac{|(\boldsymbol{f}'(\boldsymbol{X}_B)\boldsymbol{V}_B^{-1}\boldsymbol{f}(\boldsymbol{X}_B))^{-1}_{2,2}|^{1/(P-1)}}{|(\boldsymbol{f}'(\boldsymbol{X}_A)\boldsymbol{V}_A^{-1}\boldsymbol{f}(\boldsymbol{X}_A))^{-1}_{2,2}|^{1/(P-1)}}. \tag{11}$$

2.2.3 $\mathcal{I}$-optimality criterion

In contrast to the $\mathcal{D}$- and $\mathcal{D}_s$-optimality criteria, the $\mathcal{I}$-optimality criterion pays attention to the quality of the predictions. More specifically, it seeks to minimize the average prediction variance over the experimental region χ,

$$\mathcal{I}(\boldsymbol{X}) = \frac{\int_\chi \boldsymbol{f}'(\boldsymbol{w},\boldsymbol{s})(\boldsymbol{f}'(\boldsymbol{X})\boldsymbol{V}^{-1}\boldsymbol{f}(\boldsymbol{X}))^{-1}\boldsymbol{f}(\boldsymbol{w},\boldsymbol{s})d(\boldsymbol{w},\boldsymbol{s})}{\int_\chi d(\boldsymbol{w},\boldsymbol{s})}. \tag{12}$$

This integrated variance can be calculated efficiently using the expression

$$\mathcal{I}(\boldsymbol{X}) = 2^{-(H+E)}\text{tr}((\boldsymbol{f}'(\boldsymbol{X})\boldsymbol{V}^{-1}\boldsymbol{f}(\boldsymbol{X}))^{-1}\boldsymbol{M}), \tag{13}$$

where $\boldsymbol{M}$ is a $P \times P$ matrix called the moment matrix. This matrix has a very specific structure and can be derived easily for each experimental scenario, as explained by Goos and Jones (2011). A design that minimizes the average prediction variance is called an $\mathcal{I}$-optimal design. In order to compare the quality of two designs with design matrices $\boldsymbol{X}_A$ and $\boldsymbol{X}_B$, we calculate the $\mathcal{I}$-efficiency of the first design relative to the second using the expression

$$\mathcal{I}\text{-eff}(\boldsymbol{X}_A, \boldsymbol{X}_B) = 100\,\frac{\mathcal{I}(\boldsymbol{X}_B)}{\mathcal{I}(\boldsymbol{X}_A)} = 100\,\frac{\text{tr}((\boldsymbol{f}'(\boldsymbol{X}_B)\boldsymbol{V}_B^{-1}\boldsymbol{f}(\boldsymbol{X}_B))^{-1}\boldsymbol{M})}{\text{tr}((\boldsymbol{f}'(\boldsymbol{X}_A)\boldsymbol{V}_A^{-1}\boldsymbol{f}(\boldsymbol{X}_A))^{-1}\boldsymbol{M})}. \tag{14}$$

2.2.4 $\mathcal{I}_d$-optimality criterion

The $\mathcal{I}_d$-optimality criterion is based on the variance dispersion graphs suggested by Trinca and Gilmour (1999). This criterion also focuses on prediction quality. However, it does not pay attention to predictions of the response variable itself, but to how it differs from a certain response value obtained for a specific factor setting. This setting could represent the current working conditions or the expected optimum configuration, and is often located close to the center of the experimental region χ. The $\mathcal{I}_d$-optimality criterion therefore seeks to minimize the average variance of the prediction of the difference between the estimated mean response at each design point in χ and the estimated mean response at the center of χ,

$$\mathcal{I}_d(\boldsymbol{X}) = \frac{\int_\chi [\boldsymbol{f}(\boldsymbol{w},\boldsymbol{s}) - \boldsymbol{f}(\boldsymbol{0}_H,\boldsymbol{0}_E)]'(\boldsymbol{f}'(\boldsymbol{X})\boldsymbol{V}^{-1}\boldsymbol{f}(\boldsymbol{X}))^{-1}[\boldsymbol{f}(\boldsymbol{w},\boldsymbol{s}) - \boldsymbol{f}(\boldsymbol{0}_H,\boldsymbol{0}_E)]d(\boldsymbol{w},\boldsymbol{s})}{\int_\chi d(\boldsymbol{w},\boldsymbol{s})}, \tag{15}$$

where $\boldsymbol{f}(\boldsymbol{0}_H,\boldsymbol{0}_E)$ is a $P\times 1$ vector whose first element is one and all others are zero. This integrated variance can be calculated efficiently using the expression

$$\mathcal{I}_d(\boldsymbol{X}) = 2^{-(H+E)}\mathrm{tr}((\boldsymbol{f}'(\boldsymbol{X})\boldsymbol{V}^{-1}\boldsymbol{f}(\boldsymbol{X}))^{-1}\boldsymbol{M}_0), \tag{16}$$

where $\boldsymbol{M}_0$ is equal to the moment matrix $\boldsymbol{M}$, except that all its elements in the first row and the first column are equal to zero. We calculate the $\mathcal{I}_d$-efficiency of a design with design matrix $\boldsymbol{X}_A$ relative to another design with design matrix $\boldsymbol{X}_B$ as

$$\mathcal{I}_d\text{-eff}(\boldsymbol{X}_A,\boldsymbol{X}_B) = 100\,\frac{\mathcal{I}_d(\boldsymbol{X}_B)}{\mathcal{I}_d(\boldsymbol{X}_A)} = 100\,\frac{\mathrm{tr}((\boldsymbol{f}'(\boldsymbol{X}_B)\boldsymbol{V}_B^{-1}\boldsymbol{f}(\boldsymbol{X}_B))^{-1}\boldsymbol{M}_0)}{\mathrm{tr}((\boldsymbol{f}'(\boldsymbol{X}_A)\boldsymbol{V}_A^{-1}\boldsymbol{f}(\boldsymbol{X}_A))^{-1}\boldsymbol{M}_0)}. \tag{17}$$

3 Motivating examples

In order to illustrate that the grouping configuration has a strong impact on the quality of a design for a two-stratum experiment, we discuss several experiments involving three 2-level factors. Bear in mind that, throughout our exposition, we assume that the variance ratio is equal to one.

3.1 An 8-run blocked experiment for a main-effects model

We first consider an experiment involving 8 observations. However, only 4 observations can be made on a single day and at most 4 days can be used to run the experiment. The most attractive design alternatives for this experiment result from orthogonally blocking a 2^3 factorial design in 4 blocks (of size 2) and in 2 blocks (of size 4). These alternatives are shown in Table 1. The low factor levels are denoted by -1, while the high factor levels are denoted by $+1$. Observe that the first design uses as many days as possible, while the second uses as many observations per day as possible. The design with 2 blocks has a $\mathcal{D}$-efficiency equal to 88.01% relative to the alternative design with 4 blocks. The design with 2 blocks is also worse in terms of the $\mathcal{I}$-optimality criterion: its $\mathcal{I}$-efficiency is 75.00% relative to the alternative design with 4 blocks. Both designs, however, perform equally well in terms of the $\mathcal{D}_s$- and $\mathcal{I}_d$-optimality criteria. In this example, for two of the four optimality criteria, using a larger number of blocks is better.

$+1$	$+1$	-1
-1	-1	$+1$
$+1$	-1	-1
-1	$+1$	$+1$
$+1$	-1	$+1$
-1	$+1$	-1
-1	-1	-1
$+1$	$+1$	$+1$

a) 4 blocks of size 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

b) 2 blocks of size 4

Table 1: Designs for a blocked experiment involving 3 factors and 8 observations.

3.2 An 8-run split-plot experiment for a main-effects model

Now, we consider the same scenario as above, but one factor is hard to change. Again, two straightforward alternative split-plot designs exist: one involving 4 whole plots (of size 2) and one involving 2 whole plots (of size 4). Both of these designs are shown in Table 2. The levels of the hard-to-change factor are indicated in bold. The difference in the statistical efficiency between these two designs is considerable: the design with 2 whole plots is inferior to the alternative with 4 whole plots in terms of all four optimality criteria. Its $\mathcal{D}$-, $\mathcal{D}_s$-, $\mathcal{I}$- and $\mathcal{I}_d$-efficiency is 77.49%, 84.34%, 66.66% and 71.42%, respectively. Moreover, the design with only 2 whole plots does not allow for enough degrees of freedom to estimate the whole-plot variance σ_δ^2. In this example too, using a larger number of whole plots is better.

+1	+1	+1
+1	−1	−1
−1	+1	+1
−1	−1	−1
+1	−1	+1
+1	+1	−1
−1	+1	−1
−1	−1	+1

a) 4 whole plots of size 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

b) 2 whole plots of size 4

Table 2: Designs for a split-plot experiment involving 1 hard-to-change factor (bold), 2 easy-to-change factors and 8 observations.

3.3 A 12-run blocked experiment for a model with main effects and interaction effects

In the previous examples, the larger the number of groups, the better the statistical efficiency of the design. However, this is not always the case. To illustrate this, we consider an experiment with 12 observations for estimating a model involving main effects and two-factor interaction effects. For this experiment, there are also two intuitive design alternatives: a design involving 4 blocks (of size 3) and a design involving 3 blocks (of size 4). These alternatives are shown in Table 3. In this case, the design involving 4 blocks is inferior to the alternative involving 3 blocks in terms of three optimality criteria. Its $\mathcal{D}$-, $\mathcal{D}_s$- and $\mathcal{I}_d$-efficiency is 93.18%, 89.54% and 87.48%, respectively. Both designs perform equally well in terms of their $\mathcal{I}$-efficiency. As a result, in this scenario, using a smaller number of blocks is better for most of the optimality criteria.

3.4 A 12-run split-plot experiment for estimating a model with main effects and interaction effects

Finally, we consider the same scenario as in Section 3.3, but assume there is one hard-to-change factor. For this experiment, there are two intuitive alternatives: a design involving 3 whole plots (of size 4) and a design involving 4 whole plots (of size 3). In this case, the split-plot design involving 3 whole plots (of size 4) is inferior to the alternative involving 4 whole plots (of size 3) in terms of three optimality criteria. The $\mathcal{D}$-, $\mathcal{I}$- and $\mathcal{I}_d$-efficiency of the former design is 98.98%, 85.61% and 91.82%, respectively. To the contrary, the design involving 4 whole plots is inferior to the alternative with 3 whole plots in terms of the $\mathcal{D}_s$-optimality criterion: its $\mathcal{D}_s$-efficiency is 97.50%. These two designs, however, do not have the optimal grouping configuration for the experiment. The optimal design is composed of 2 whole plots of size 2 and 2 whole plots of size 4. This design is 2.72% more $\mathcal{D}$-efficient, 2.17% more $\mathcal{I}$-efficient and 3.07% more $\mathcal{I}_d$-efficient than the

alternative with 4 homogeneous whole plots of size 3. This design is also 0.98% more $\mathcal{D}_s$-efficient than the alternative with 3 homogeneous whole plots of size 4. The three different designs are shown in Table 4.

These four examples show that the grouping configuration has a considerable impact on the statistical efficiency of a design. Furthermore, the examples show that the optimal grouping configuration is not always obvious, and that it depends on the characteristics of the experiment: the type of two-stratum experiment, the model to be estimated and the optimality criterion considered.

-1	-1	-1
-1	$+1$	$+1$
$+1$	$+1$	-1
$+1$	$+1$	$+1$
-1	-1	$+1$
-1	$+1$	-1
-1	-1	-1
$+1$	$+1$	-1
$+1$	-1	$+1$
$+1$	$+1$	$+1$
-1	-1	$+1$
$+1$	-1	-1

a) 4 blocks of size 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	$+1$

b) 3 blocks of size 4

Table 3: Designs for a blocked experiment involving 3 factors and 12 observations.

$\mathbf{+1}$	-1	$+1$
$\mathbf{+1}$	$+1$	$+1$
$\mathbf{+1}$	-1	-1
$\mathbf{-1}$	-1	$+1$
$\mathbf{-1}$	$+1$	-1
$\mathbf{-1}$	$+1$	$+1$
$\mathbf{+1}$	$+1$	$+1$
$\mathbf{+1}$	$+1$	-1
$\mathbf{+1}$	-1	$+1$
$\mathbf{-1}$	-1	-1
$\mathbf{-1}$	$+1$	$+1$
$\mathbf{-1}$	-1	$+1$

a) 4 whole plots of size 3

$\mathbf{+1}$	$+1$	-1
$\mathbf{+1}$	-1	-1
$\mathbf{+1}$	$+1$	$+1$
$\mathbf{+1}$	$+1$	-1
$\mathbf{-1}$	$+1$	$+1$
$\mathbf{-1}$	$+1$	-1
$\mathbf{-1}$	-1	-1
$\mathbf{-1}$	-1	$+1$
$\mathbf{+1}$	$+1$	-1
$\mathbf{+1}$	-1	-1
$\mathbf{+1}$	$+1$	$+1$
$\mathbf{+1}$	-1	$+1$

b) 3 whole plots of size 4

$\mathbf{-1}$	-1	$+1$
$\mathbf{-1}$	-1	-1
$\mathbf{-1}$	$+1$	$+1$
$\mathbf{-1}$	$+1$	-1
$\mathbf{+1}$	-1	$+1$
$\mathbf{+1}$	$+1$	-1
$\mathbf{-1}$	-1	-1
$\mathbf{-1}$	$+1$	$+1$
$\mathbf{-1}$	-1	$+1$
$\mathbf{-1}$	$+1$	-1
$\mathbf{+1}$	-1	-1
$\mathbf{+1}$	$+1$	$+1$

c) 4 whole plots of different sizes

Table 4: Designs for a split-plot experiment involving 1 hard-to-change factor (bold), 2 easy-to-change factors and 12 observations.

4 Literature review

A large portion of the literature on the generation of designs for two-stratum experiments focuses on combinatorial constructions of fractional factorial designs. For example, Bisgaard (1994), Sun et al. (1997), Sitter et al. (1997), Zhang and Park (2000) and Cheng and Wu (2002) discuss blocking schemes for fractional factorial designs considering the minimum aberration criterion. Similar schemes are presented by Huang et al. (1998), Bingham and Sitter (1999) and Bingham et al. (2004, 2005) for split-plot experiments. Fractional factorial designs for strip-plots experiments and more complex configurations of hard-to-change factors are discussed by Miller (1997), Mee and Bates (1998) and Butler (2004).

The literature previously referenced focuses on how to arrange traditional experimental designs in a two-stratum fashion. Unfortunately, these designs cannot always be used since they exist only for a limited set of experimental scenarios. Algorithmic design generation approaches are more flexible since they can construct designs tailored to the characteristics of an experiment. Many of the algorithms for the design of two-stratum experiments are based on the traditional *point-exchange algorithm* proposed by Fedorov (1972), such as the algorithms proposed by Atkinson and Donev (1989), Cook and Nachtsheim (1989) and Goos and Vandebroek (2001b) for the design of blocked experiments, and the algorithms proposed by Goos and Vandebroek (2001a, 2003, 2004) and Smucker et al. (2012) for the design of split-plot experiments. A drawback of point-exchange algorithms is that they make use of a candidate set of design points, which defines the possible factor-level combinations. Point-exchange algorithms iteratively explore the candidate set in order to find the best combination of points to compose the design. The size of the candidate set grows exponentially with the number of factors. For this reason, when the number of factors is relatively large, exploring or even constructing such a set may become computationally intractable.

The coordinate-exchange algorithm (CEA) proposed by Meyer and Nachtsheim (1995) was the first alternative approach to the point-exchange algorithm. Instead of using a candidate set of design points, this algorithm iteratively improves the design points one coordinate or factor at a time. Due to this characteristic, this approach is more efficient for the design of experiments involving relatively large numbers of factors. More recently, several algorithms based on the CEA have been proposed for the design of two-stratum experiments. Examples are the algorithms proposed by Jones and Goos (2007), Sambo et al. (2014) and Mylona et al. (2014) for the design of split-plot experiments. Similar algorithms have also been proposed for the design of experiments involving more complex strata configurations. Examples are the algorithm proposed by Jones and Goos (2009) for the design of split-split-plot experiments, the algorithm proposed by Arnouts et al. (2010) for the design of strip-plot experiments and that proposed by Arnouts and Goos (2012, 2015) for the design of staggered-level experiments.

The sequential algorithm proposed by Trinca and Gilmour (2001) for the design of general multi-stratum experiments deserves special attention due to its flexibility: it allows the experimenter to specify any stratum configuration. Another advantage is that the method is independent of the ratios of the variance components. The algorithm generates a design by sequentially choosing the factor levels in each stratum such that they are as orthogonal as possible with respect to higher strata. Nevertheless, some of the designs generated by this algorithm are not as statistically efficient as the designs generated by specialized algorithms (see, for example, Jones and Goos (2007)). This inspired Trinca and Gilmour (2015) to improve their original sequential algorithm. To this end, they make use of exchange algorithms in order to further improve the statistical efficiency of the designs generated. This new feature makes the algorithm more computationally intensive. However, it allows to generate better designs that are more robust in terms of several optimality criteria.

Despite the benefits of considering a flexible grouping configuration, the generation of optimal designs where the number of groups and the sizes of the groups are not predefined has received little attention. The algorithms described by Goos and Vandebroek (2004) and Kessels et al. (2008) are the only ones for two-stratum experiments with these flexible characteristics. They generate designs by not only optimizing

the settings of the factor levels, but also the grouping configuration. These algorithms, however, only allow to impose restrictions on the number of groups but not on the group sizes. Additionally, these algorithms are point-exchange algorithms. For this reason, their execution time exponentially increases with the number of factors and observations.

The impact of the grouping configuration on the statistical efficiency of optimal designs for two-stratum experiments has also received little attention. Goos (2002) carried out computational simulations to study the benefits of increasing the number of whole plots in the generation of optimal split-plot designs. In the context of conjoint experiments, Kessels et al. (2008) showed how the grouping configuration of optimal blocked designs depends on the experimental scenario.

The computational simulations carried out by Goos (2002), Goos and Vandebroek (2004) and Kessels et al. (2008) focus exclusively on $\mathcal{D}$-optimal designs, and ignore criteria that are more suitable if the goal of the experiment is to make predictions. In this paper, we also study the impact of the grouping configurations considering the $\mathcal{D}_s$-, $\mathcal{I}$- and $\mathcal{I}_d$-optimality criteria.

5 Algorithmic framework

In this section, we explain our algorithmic framework for the optimal design of two-stratum experiments in case the number of groups and the size of the groups are limited only by upper bounds. Although we focus on the generation of designs for two-stratum experiments, the framework can be adapted to handle larger numbers of strata.

The algorithmic framework is based on a multi-level local search approach: it is composed of two optimization layers, one for each stratum. These optimization layers are superimposed, in the sense that the higher layer executes the lower one as a part of its optimization process. Each layer improves the quality of the design by making small structural changes to its stratum. The quality of a design is measured by the value of the optimality criterion considered. In order to calculate this value, it is necessary to specify the variance ratio $\sigma_\delta^2/\sigma_\varepsilon^2$ as an input parameter. In the remaining sections, we first describe the low-level optimization layer and we then move to the high-level layer.

5.1 Low-level optimization layer: improving the factor levels

The low-level layer optimizes the factor levels of the experimental observations. This is done by considering all observations that belong to a single group, say group k. The factor levels of these observations are improved using a modified version of the CEA proposed by Jones and Goos (2007). This algorithm involves two phases. The first phase improves the levels of the hard-to-change factors that are constant within group k. The second phase improves the levels of the easy-to-change factors for each individual observation within group k. The pseudocode of this optimization layer is shown in Algorithm 1.

The algorithm requires an initial design matrix $\boldsymbol{X}$ and the index of the group k as input parameters. The function $\mathcal{Q}(\boldsymbol{X})$ calculates the quality measure of the design. This quality measure is either the $\mathcal{D}$-optimality criterion value or the negative of the $\mathcal{D}_s$-, $\mathcal{I}$- or $\mathcal{I}_d$-optimality criteria values. In this way, the algorithm is capable of handling the generation of an optimal design as a maximization problem regardless of the optimality criterion. The variable $\mathcal{Q}^0$ stores the quality of the initial design, and the variable $\mathcal{Q}^*$ stores the quality of the best design found during the execution of the algorithm. The optimization procedure implements a first-improvement strategy. In other words, the algorithm applies a beneficial exchange as soon as it is discovered. The first phase of the algorithm iterates over the H hard-to-change factors in the design (lines 2 to 10 in the pseudocode). For each hard-to-change factor j, the algorithm aims at finding the best level to be assigned to the entire group. To this end, it iterates over the possible factor levels represented by $1, \ldots, h_j$ (lines 5 to

Algorithm 1: Improving the factor-level combinations within a group (`improve_observations`).

Input: The design matrix $\boldsymbol{X}$ and the group index k

```
1  Q^0 ← Q(X)
   // Improve the levels of the hard-to-change factors
2  for j ← 1 to H do
3      l* ← w_kj
4      Q* ← Q(X)
5      for l ← 1 to h_j do
6          w_kj ← l
7          if Q(X) > Q* then
8              Q* ← Q(X)
9              l* ← l
10     w_kj ← l*
   // Improve the levels of the easy-to-change factors
11 for i ← 1 to N_k do
12     for j ← 1 to E do
13         l* ← s_kij
14         Q* ← Q(X)
15         for l ← 1 to e_j do
16             s_kij ← l
17             if Q(X) > Q* then
18                 Q* ← Q(X)
19                 l* ← l
20         s_kij ← l*
21 if Q(X) > Q^0 then
22     improve_observations(X, k)
```

9 in the pseudocode) and assigns the one that produces the best value of $\mathcal{Q}(\boldsymbol{X})$ (line 10 in the pseudocode). The second phase of Algorithm 1 follows a similar strategy: it iterates over the N_k observations in group k (lines 11 to 20 in the pseudocode) and aims at improving the levels of the easy-to-change factors. It does so by iterating over the possible easy-to-change factor levels represented by $1, \ldots, e_j$ (lines 15 to 19 in the pseudocode) and by assigning the level that produces the design with the best quality (line 20 in the pseudocode).

5.2 High-level optimization layer: improving the grouping configuration

For any given grouping configuration, the low-level optimization layer improves the factor settings of the observations within a group. This low-level layer is executed by the higher one, which optimizes the grouping configuration of the design (i.e., the number of groups and the size of each group). The core component of this high-level layer is a *resizing operator*, which modifies the sizes of the groups in the design. This operator has the indices k_1 and k_2 of two groups as input parameters, and works as follows. First, it randomly removes one observation i from group k_1, thereby decreasing the size of group k_1 by one unit. Next, the size of group k_2 is increased by including a new observation with the same levels of the easy-to-change factors as observation i, and with the same levels of the hard-to-change factors as the rest of observations within group k_2.

Table 5 shows an example of a modification performed by the resizing operator to two consecutive whole plots in a split-plot design. The first column (printed in bold) contains the levels of the hard-to-change factor,

while the next columns contain the levels of the easy-to-change factors. The size of the upper whole plot is decreased by deleting the second observation. The size of the lower whole plot is then increased by adding an observation with the same levels of the easy-to-change factors. Observe that the resizing operator, in principle, can be understood as an operator that modifies two disjoints sets: it removes a random element from one set, and inserts an element with similar properties in the other set. In this sense, we can consider each subdesign matrix $\boldsymbol{X}_k$ (with $1 \leq k \leq K$) as a set of N_k observations $\{(\boldsymbol{w}_k, \boldsymbol{s}_{k1}), \ldots, (\boldsymbol{w}_k, \boldsymbol{s}_{kN_k})\}$. Similarly, we can consider the design matrix $\boldsymbol{X}$ as a set of K subdesign matrices $\{\boldsymbol{X}_1, \ldots, \boldsymbol{X}_K\}$. This interpretation is helpful to give a clear explanation of the algorithm, and is later used in the pseudocode for this optimization layer.

⋮	⋮	⋮
+1	−1	+1
+1	*+1*	*+1*
+1	−1	−1
+1	+1	−1
−1	+1	−1
−1	−1	+1
−1	+1	−1
−1	−1	−1
⋮	⋮	⋮

a) Before

⋮	⋮	⋮
+1	−1	+1
+1	−1	−1
+1	+1	−1
−1	+1	−1
−1	*+1*	*+1*
−1	−1	+1
−1	+1	−1
−1	−1	−1
⋮	⋮	⋮

b) After

Table 5: Application of one iteration of the resizing operator.

The high-level optimization layer iterates over all possible pairs of groups in the design. The resizing operator is executed several times in order to perform local changes to the grouping configuration. Every time a pair of groups is modified by the operator, the low-level optimization layer (shown in Algorithm 1) is executed in order to improve the factor-level combinations they contain. The number of times T the resizing operator is applied to a given pair of groups determines the strength of the high-level optimization. The algorithm starts by considering $T = 1$ and evaluating the designs produced by small modifications only. When the optimization layer gets trapped in a locally optimal design (i.e., when it cannot further improve the quality of the current design), the value of T is increased in order to perform a more thorough exploration (i.e., to consider larger modifications). The value of T is reset to one when a stronger modification produces a design with better quality and the algorithm escapes from the local optimum.

The exploration strategy previously explained resembles that of the variable neighborhood search (VNS) algorithm proposed by Mladenović and Hansen (1997). The VNS is a local-search-based metaheuristic that has been successfully applied to several combinatorial optimization problems (Hansen and Mladenović, 2003). A VNS algorithm receives an initial solution as an input, and attempts to improve it by exploring one of its *neighborhoods*. A neighborhood of a solution is defined as the set of all the solutions produced by applying a specific type of local change to it. When the VNS cannot find a better solution within a specific neighborhood (i.e., when the current solution is a local optimum with respect to the neighborhood), the algorithm switches to a different neighborhood in order to escape from the local optimum. This new neighborhood is usually more complex than the previous one, and attempts to achieve a more thorough exploration of the solution space. By increasing the value of T, the high-level optimization layer of our algorithm follows a similar strategy.

The pseudocode of the high-level optimization layer is shown in Algorithm 2. The algorithm requires the design matrix $\boldsymbol{X}$ of an initial design, the maximum number of groups, G, and the maximum number of

Algorithm 2: Improving the grouping configuration (`improve_groups`).

Input: The design matrix $\boldsymbol{X}$ of an initial design, the maximum number of groups, G, and the maximum number of observations per group, S

$\boldsymbol{X}^0 \leftarrow \boldsymbol{X}$
$T \leftarrow 1$
while $T \leq \max(N_k) \mid 1 \leq k \leq K$ **do**
  $\boldsymbol{X}^* \leftarrow \boldsymbol{X}$
  if $((\forall k \mid 1 \leq k \leq K \ : \ N_k \neq 0) \wedge K < G)$ **then**
    $K \leftarrow K + 1$
    $\boldsymbol{X} \leftarrow \boldsymbol{X} \cup \boldsymbol{\Omega}$
  for $k_1 \leftarrow 1$ **to** K **do**
    for $k_2 \leftarrow 1$ **to** K **do**
      if $k_1 \neq k_2 \wedge N_{k_1} \geq T \wedge N_{k_2} + T \leq S$ **then**
        for $t \leftarrow 1$ **to** T **do**
          Randomly select $i \mid 1 \leq i \leq N_{k_1}$
          $\boldsymbol{X}_{k_2} \leftarrow \boldsymbol{X}_{k_2} \cup (\boldsymbol{w}_{k_2}, \boldsymbol{s}_{k_1 i})$
          $N_{k_2} \leftarrow N_{k_2} + 1$
          $\boldsymbol{X}_{k_1} \leftarrow \boldsymbol{X}_{k_1} \setminus (\boldsymbol{w}_{k_1}, \boldsymbol{s}_{k_1 i})$
          $N_{k_1} \leftarrow N_{k_1} - 1$
        `improve_observations`$(\boldsymbol{X}, k_1)$
        `improve_observations`$(\boldsymbol{X}, k_2)$
        if $\mathcal{Q}(\boldsymbol{X}) > \mathcal{Q}(\boldsymbol{X}^*)$ **then**
          $\boldsymbol{X}^* \leftarrow \boldsymbol{X}$
          if $((\forall k \mid 1 \leq k \leq K \ : \ N_k \neq 0) \wedge K < G)$ **then**
            $K \leftarrow K + 1$
            $\boldsymbol{X} \leftarrow \boldsymbol{X} \cup \boldsymbol{\Omega}$
        else
          $\boldsymbol{X} \leftarrow \boldsymbol{X}^*$
  for $k \leftarrow K$ **to** 1 **do**
    if $N_k = 0$ **then**
      $\boldsymbol{X} \leftarrow \boldsymbol{X} \setminus \boldsymbol{X}_k$
      $K \leftarrow K - 1$
  if $\mathcal{Q}(\boldsymbol{X}^*) > \mathcal{Q}(\boldsymbol{X}^0)$ **then**
    $\boldsymbol{X}^0 \leftarrow \boldsymbol{X}^*$
    $T \leftarrow 1$
  else
    $T \leftarrow T + 1$
  $\boldsymbol{X} \leftarrow \boldsymbol{X}^0$

observations per group, S, as input parameters. The variable $\boldsymbol{X}^0$ stores the initial design, while the variable $\boldsymbol{X}^*$ stores the best design found during the execution of the algorithm. The variable T is initialized to one (line 2 in the pseudocode) and represents the number of times the resize operator is applied to each pair of groups. In case the number of groups that compose the initial design is smaller than G, the algorithm extends the grouping configuration by creating a dummy group $\boldsymbol{\Omega}$ that is empty (lines 5 to 7 in the pseudocode). This allows the algorithm to evaluate the possibility of adding a group to the design. The algorithm then iterates over every pair of group indices k_1 and k_2 (lines 8 to 25 in the pseudocode), and evaluates the designs resulting from modifying the size of the groups. This modification comprises T iterations of the resizing operator (lines 11 to 16 in the pseudocode) and the execution of the low-level optimization layer in order to improve the factor levels of both groups under consideration (lines 17 and 18 in the pseudocode). Note that this modification is evaluated only when the application of the resizing operator does not result in a violation of either the maximum number of groups or the maximum number of observations per group (line 10 in the pseudocode). If the resulting design is better than the original one, it is used to continue the optimization process (lines 19 to 23 in the pseudocode); otherwise, the design is discarded (lines 24 and 25 in the pseudocode). After evaluating the modification of all possible pairs of groups, the algorithm checks whether the resulting design has any empty groups. If so, they are eliminated from the grouping configuration (lines 26 to 29 in the pseudocode). If the algorithm was able to find a better design during the exploration process, the value of T is reset to one (lines 30 to 32 in the pseudocode). Otherwise, the value of T is increased to attempt to escape from the locally optimal design. The execution of the algorithm stops when the strongest possible modification cannot improve the quality of the design. This happens when the value of T exceeds the size of the largest group in the design (line 3 in the pseudocode). In the appendix, we study the influence of the maximum value of T on the performance of the algorithmic framework.

The initial design required by the optimization layer is randomly generated. This design is constructed in such a way that the experimental constraints are satisfied. This means that the number of groups is lower than or equal to G, and the size of each group is lower than or equal to S. Random levels are assigned to the hard-to-change factors for each of the groups in the design. Similarly, random levels are assigned to the easy-to-change factors for each observation. We recommend executing the algorithmic framework using a multi-start strategy. In other words, we suggest executing the algorithm several times, starting from different initial random designs, and selecting the best design generated. This strategy is common to most of the algorithms for the generation of optimal designs and has become a standard practice.

5.3 Analysis and theoretical comparison

The algorithmic framework outlined in Sections 5.1 and 5.2 generalizes the coordinate-based approach proposed by Jones and Goos (2007). In the special case where the number of observations is equal to the maximum number of groups multiplied by the maximum group size, the high-level optimization layer is not executed and our framework reduces to the algorithm that they propose. Our algorithmic framework has two major advantages over the point-exchange algorithm proposed by Goos and Vandebroek (2004). First, it does not require specifying a set of candidate points. Second, its execution times is proportional to $K(K-1)(\sum_{i=1}^{H} h_i + N/K \sum_{i=1}^{E} e_i)$ (assuming that the observations are evenly distributed among the groups and $N_k \approx N/K$, for $1 \leq k \leq K$), while the execution time of the point-exchange algorithm is proportional to $(2+K)N \prod_{i=1}^{H} h_i \prod_{i=1}^{E} e_i$. This difference in execution time increases with the number of factors.

The application of the resizing operator as a core component of the framework has several advantages. First, it allows the specification of how thorough the search for an optimal design is. This is done by establishing a maximum value for the number of times T the operator is applied to a pair of groups. Second, it allows to evaluate a wide range of structural changes to the design. Depending on the pair of groups (k_1, k_2) the operator is applied to and the value of T, the framework evaluates quite different changes to the grouping configuration. For example, if $T = N_{k_1}$, then groups k_1 and k_2 are merged into a single group by

the resizing operator. Conversely, if group k_2 is empty and $T < N_{k_1}$, then group k_1 is split into two different groups by the resizing operator (where the size of the newly created group equals T). These kinds of changes are not evaluated by the point-exchange algorithm of Goos and Vandebroek (2004). Their algorithm only evaluates changes involving two different factor-level combinations. More specifically, it considers

- Replacing a design point in a group with a point from the candidate set;
- Removing a design point from one group and adding a design point from the candidate set to a new group;
- Removing a design point from one group and adding a design point from the candidate set to another existing group.

Due to this limited set of modifications, the exploration performed by the algorithm of Goos and Vandebroek (2004) is substantially less thorough than that performed by our algorithmic framework. Additionally, in its execution, the point-exchange algorithm uses a best-improvement selection strategy. In other words, at each iteration, the algorithm evaluates all possible changes and eventually implements the change that produces the design with the best quality. In contrast, our algorithmic framework uses a first-improvement selection strategy: it applies a beneficial change as soon as it has been identified. The computational simulations reported by Palhazi Cuervo et al. (2016) suggest that, in general, using a first-improvement strategy is considerably better: it leads to faster execution times without affecting the quality of the designs generated.

5.4 Extension to more general multi-stratum experiments

In order to adapt the algorithmic framework for the generation of optimal designs for more complex multi-stratum experiments, it is necessary to implement some modifications. It is important to adapt the framework in order to handle more general constraints imposed on the factor-level configurations. These constraints define which observations must share the same levels for some of the factors due to the multi-stratum structure. In this case, the framework should include as many optimization layers as there are nested strata in the experiment. These optimization layers should be superimposed. In other words, the layer that optimizes the grouping configuration of one stratum should execute the layer that optimizes the grouping configuration of the next lower nested stratum. Additionally, if the experiment considers different grouping configurations within the same stratum (as, for example, in staggered-level experiments), one optimization procedure should be implemented for each grouping configuration. These procedures should act sequentially at the same optimization level when improving their stratum. This strategy is required because the grouping configurations are then not nested.

6 Computational simulations

In this section, we discuss the results of a computational simulation carried out with two purposes. The first goal is to further analyze how the grouping configuration of an optimal design depends on the characteristics of the experiment. More specifically, the goal is to investigate how it depends on:

- The experimental factors involved;
- The type of two-stratum experiment considered;
- The model to be estimated;
- The optimality criterion used.

The second goal is to study the effects of the experimental constraints (the maximum number of groups and the maximum group size) on the statistical efficiency of the designs. In order to do so, we executed the algorithmic framework described in Section 5 in 10 different experimental scenarios. These scenarios involve two different sets of experimental factors:

1. A set composed of two 2-level categorical factors, and two 3-level categorical factors.

2. A set composed of two continuous factors, one 2-level categorical factor, and one 4-level categorical factor.

We chose these heterogeneous sets of factors because they are very different from those studied in papers on classical (fractional) factorial designs, even though it is common to find such factor configurations in real-life experiments. The best grouping configurations for these sets of heterogeneous factors are therefore unknown. For each set of factors, we considered a model involving main effects only and a model involving main effects and two-factor interaction effects. For the second set of factors, we also considered a model involving main effects, interaction effects and quadratic effects. For scenarios where a main-effects model was to be estimated, we set the total number of observations to 12. For scenarios where a more complex model was to be estimated, we set the total number of observations to 24.

For each combination of a factor set and a model, we generated both blocked and split-plot experiments. For the latter, we considered two factors to be hard to change and two factors to be easy to change. Table 6 summarizes the information about the 10 experimental scenarios. For each scenario, the table shows the type of two-stratum experiment, the configuration of factors, the model to be estimated, the total number of observations and the corresponding identifier. In the second column, the factors represented by a number x are categorical x-level factors, while the factors represented by the capital letter C are continuous. In the third column, the models to be estimated are represented using the initial letters of the effects they involve. For instance, M + I + Q represents a model involving main effects, interaction effects and quadratic effects. For the split-plot experiments (the last five rows in the table), the hard-to-change factors are indicated in bold.

Table 6: Scenarios considered in the computational simulations.

Type	Factors	Model to be estimated	Num. of obs.	Identifier
Blocked	2 3 2 3	M	12	Block-1-M
		M + I	24	Block-1-MI
	C 2 C 4	M	12	Block-2-M
		M + I	24	Block-2-MI
		M + I + Q	24	Block-2-MIQ
Split-plot	**2 3** 2 3	M	12	Split-1-M
		M + I	24	Split-1-MI
	C 2 C 4	M	12	Split-2-M
		M + I	24	Split-2-MI
		M + I + Q	24	Split-2-MIQ

We generated optimal designs for each of the scenarios shown in Table 6 considering different upper bounds for the number of groups and for the size of the groups. For each scenario, we evaluated all feasible combinations of upper bounds in the set $\{1, \ldots, 10\} \times \{1, \ldots, 10\}$ and generated $\mathcal{D}$-, $\mathcal{D}_s$-, $\mathcal{I}$- and $\mathcal{I}_d$-optimal designs. For each set of optimal designs (generated for a given scenario and a given optimality criterion), we identified the design with the best quality. The corresponding optimality criterion value of this design was then used as the basis for calculating the statistical efficiencies of the other designs. In other words, we calculated the statistical efficiencies of the designs with respect to the one that has the optimal grouping configuration (with at most 10 groups and at most 10 observations per group). This allowed us to determine the extent to which imposing tighter experimental constraints affects the quality of the designs generated.

The algorithmic framework was executed using 2000 restarts. Even though the framework usually finds a good-quality design using a much smaller number of restarts, we used the number 2000 to ensure a high probability that it finds the best possible design. In order to visualize the statistical efficiencies for each set of optimal designs generated (for each of the 10 experimental scenarios), we make use of heat maps. A heat map is a graphical representation of a table where the individual values are shown as colors instead of numbers, allowing for a more visual comparison and analysis. In the following sections, we provide a detailed explanation of the heat maps and we discuss the sets of optimal designs generated. We first focus on blocked experiments. Next, we proceed with split-plot experiments.

6.1 Optimal designs for blocked experiments

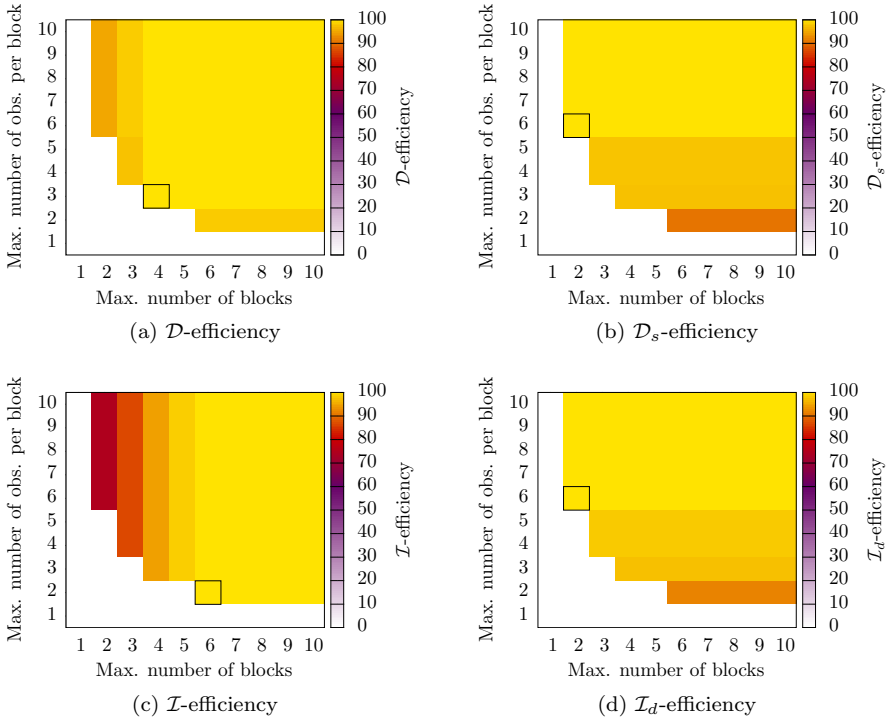
6.1.1 Main-effects models

Figures 1 and 2 show the heat maps representing the quality of the designs generated for scenarios Block-1-M and Block-2-M, respectively. These scenarios correspond to blocked experiments involving main-effects models and a total number of observations equal to 12. First, we focus on Figure 1 in order to explain how to interpret the heat maps. The heat map in Figure 1(a) shows the quality of the designs generated using the $\mathcal{D}$-optimality criterion. The heat maps in Figures 1(b), 1(c) and 1(d) show the quality of the designs generated using the $\mathcal{D}_s$-, $\mathcal{I}$- and $\mathcal{I}_d$-optimality criteria, respectively. In each heat map, the vertical axis shows the maximum number of observations per group and the horizontal axis shows the maximum number of groups. The closer a coordinate to the origin $(0, 0)$, the more restrictive the experimental constraints. Conversely, the closer the coordinate to the upper-right corner $(10, 10)$, the more flexible the combination of upper bounds. A legend explains the meaning of the colors representing the statistical efficiencies of the designs generated. The color white indicates that the corresponding combination of upper bounds does not allow the generation of a feasible design. This is either because it does not allow to form a design with the number of observations specified, or because the number of groups is too small to estimate the corresponding model. Designs that are at most 60% efficient are represented using different purple tones. The color scale then changes to red (designs that are about 80% efficient), orange (designs that are about 90% efficient) and yellow for the designs with the best optimality criterion values. The combination of upper bounds that leads to the design with the best optimality value, and therefore with the optimal grouping configuration, is indicated using a black square.

Figure 1(a) shows that, for scenario Block-1-M, the design with the best $\mathcal{D}$-optimality criterion value involves 4 blocks of size 3. Note that relaxing the experimental constraints (i.e., allowing larger numbers of blocks and larger block sizes) does not lead to better designs: the algorithm always converges to a design with 4 blocks of size 3. As a result, the framework is able to identify the blocking structure that complies with the experimental constraints and that leads to the most efficient design. This causes many of the upper bound combinations to lead to the same statistical efficiency, and therefore, the heat map to show large areas with the same color. Figure 1(a) also shows that imposing very restrictive constraints has a low impact on the $\mathcal{D}$-efficiency of the designs generated. The design with the lowest quality is generated when allowing only 2 blocks of size 6. This design is around 95% $\mathcal{D}$-efficient relative to the $\mathcal{D}$-optimal design with the best blocking configuration.

Figure 1(b) shows a different pattern. The design involving 2 blocks of size 6 is actually the design with the best $\mathcal{D}_s$-optimality criterion value. Moreover, imposing constraints on the size of the blocks has a considerable impact on the $\mathcal{D}_s$-efficiency of the designs. The design composed of 6 blocks of size 2 is only 90.10% $\mathcal{D}_s$-efficient relative to the $\mathcal{D}_s$-optimal design with 2 blocks of size 6. In contrast, Figure 1(c) shows that the design with 6 blocks of size 2 is the design with the best $\mathcal{I}$-optimality criterion value. Note that, by imposing a small number of blocks, the $\mathcal{I}$-efficiency of the designs diminishes considerably. Designs composed of 2 blocks are only around 75% $\mathcal{I}$-efficient relative to the $\mathcal{I}$-optimal design with 6 blocks of size 2. Finally, Figure 1(d) shows a pattern very similar to that shown in Figure 1(b). The design with the best

$\mathcal{I}_d$-optimality criterion value is also composed of 2 blocks of size 6, and imposing constraints on the size of the blocks has also a considerable impact on the $\mathcal{I}_d$-efficiency of the designs.



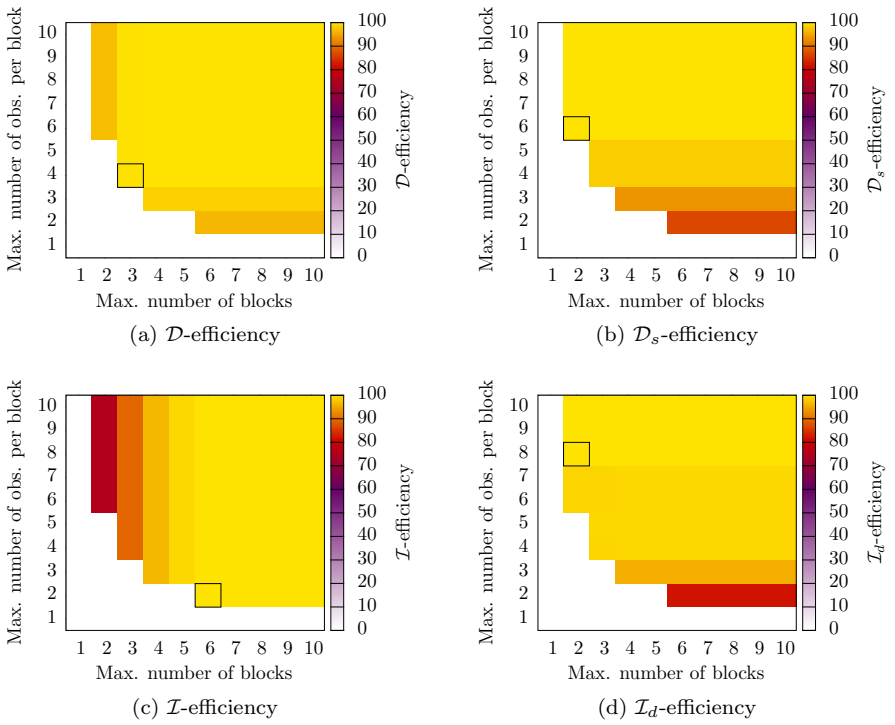
(a) $\mathcal{D}$-efficiency (b) $\mathcal{D}_s$-efficiency

(c) $\mathcal{I}$-efficiency (d) $\mathcal{I}_d$-efficiency

Figure 1: Efficiencies of the designs generated for scenario Block-1-M.

The heat maps in Figure 2 show some patterns similar to those in Figure 1 For scenario Block-2-M, the design with the best $\mathcal{D}_s$-optimality criterion value also involves 2 blocks of size 6 (see Figure 2(b)). Additionally, the design involving 6 blocks of size 2 is also the design with the best $\mathcal{I}$-optimality criterion value (see Figure 2(c)). Nevertheless, there are also a few noticeable differences between both sets of heat maps. Figure 2(a) shows that the $\mathcal{D}$-optimal design with the best blocking configuration has 3 blocks of size 4. This configuration differs from that of the design with the best $\mathcal{D}$-optimality criterion value in scenario Block-1-M. Moreover, Figure 2(d) shows that the $\mathcal{I}_d$-optimal design with the best blocking configuration involves 2 blocks with a maximum size of 8 (the design has one block of 8 observations and one block of 4 observations). This design is around 1% more $\mathcal{I}_d$-efficient than the homogeneous design involving 2 blocks of size 6. This is due to the fact that the homogeneous design is not orthogonally blocked, as opposed to the design with one block of size 8 and one block of size 4.

Figures 1 and 2 show that, when generating designs for blocked experiments involving main-effects models, $\mathcal{I}$-optimal designs seem to have larger numbers of blocks than $\mathcal{D}$-, $\mathcal{D}_s$- and $\mathcal{I}_d$-optimal designs. Imposing small numbers of blocks has therefore a larger detrimental effect on the $\mathcal{I}$-efficiency of the designs. Additionally, $\mathcal{D}_s$- and $\mathcal{I}_d$-optimal designs show similar characteristics. In other words, both optimality criteria lead to designs with similar blocking configurations. Furthermore, imposing constraints on the size of the blocks affects both optimality criteria in a similar way.



(a) $\mathcal{D}$-efficiency (b) $\mathcal{D}_s$-efficiency

(c) $\mathcal{I}$-efficiency (d) $\mathcal{I}_d$-efficiency

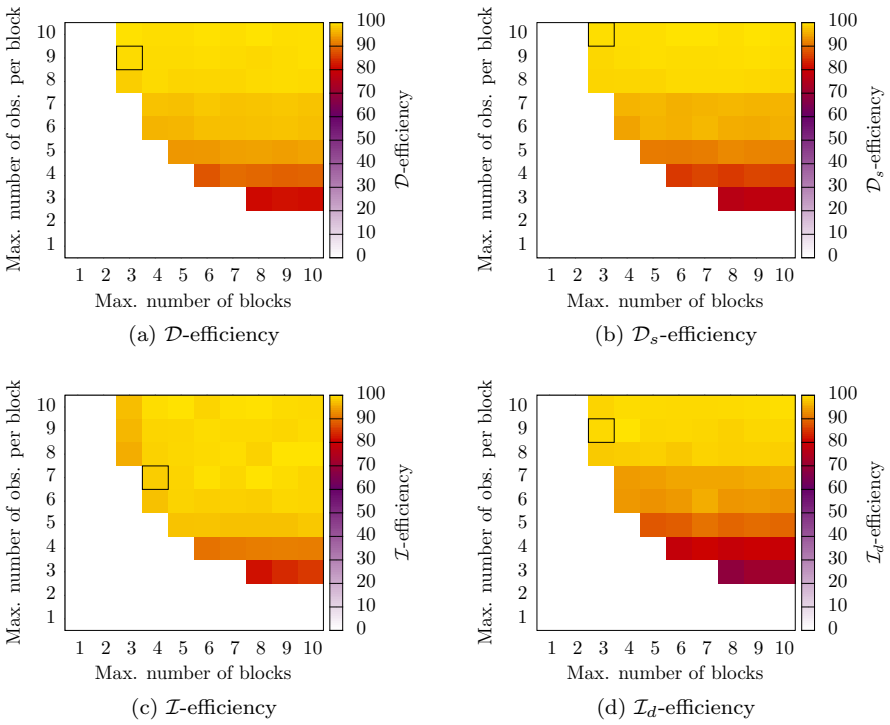
Figure 2: Efficiencies of the designs generated for scenario Block-2-M.

6.1.2 Models with interaction effects and quadratic effects

Figures 3, 4 and 5 show the heat maps for scenarios Block-1-MI, Block-2-MI and Block-2-MIQ, respectively. Scenarios Block-1-MI and Block-2-MI correspond to blocked experiments for estimating models with main effects and interaction effects, and involve a total number of observations equal to 24. Scenario Block-2-MIQ is similar to scenario Block-2-MI, but the model also involves quadratic effects.

The heat maps in Figures 3, 4 and 5 show color patterns that are less uniform than those in Figures 1 and 2. This is due to the fact that, when considering flexible experimental constraints for models other than main-effects models, the algorithm does not always find the design with the optimal blocking configuration. The suboptimal designs produced, however, show blocking configurations very similar to the optimal ones, and their statistical efficiencies differ by less than 2%. This phenomenon indicates that generating optimal designs for blocked experiments involving models with interaction effects and quadratic effects is substantially harder (despite the thorough exploration of the solution space performed by the algorithmic framework). Nevertheless, the heat maps in Figures 3, 4 and 5 clearly show how the experimental constraints affect the quality of the designs generated.

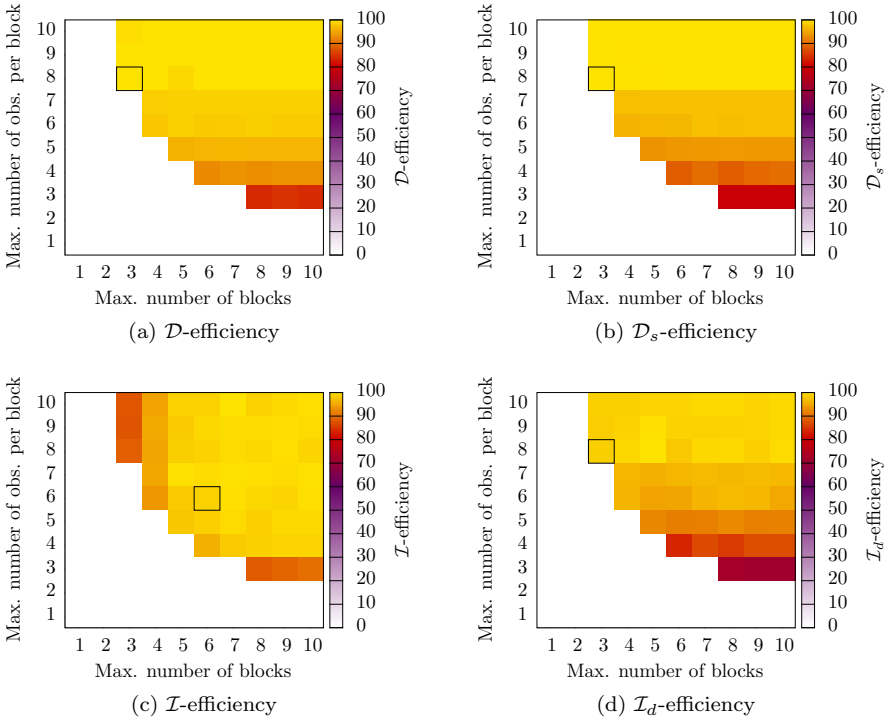
The heat maps in Figures 3 and 4 show that, when considering blocked experiments involving models with interaction effects, the best blocking configurations involve few blocks of relatively large sizes. Figures 3(a), 3(b) and 3(d) show very similar patterns. For scenario Block-1-MI, the designs with the best $\mathcal{D}$-, $\mathcal{D}_s$- and $\mathcal{I}_d$-optimality criterion values involve 3 blocks. Moreover, imposing constraints on the size of the blocks has a similar impact on these three optimality criteria. For example, a design involving 8 blocks of size 3 is only 68.02% $\mathcal{I}_d$-efficient relative to the $\mathcal{I}_d$-optimal design with the best blocking configuration (involving 3 blocks). According to Figure 3(c), the design with the best $\mathcal{I}$-optimality criterion value involves 4 blocks. This design is around 5% more $\mathcal{I}$-efficient than designs involving only 3 blocks.



(a) $\mathcal{D}$-efficiency (b) $\mathcal{D}_s$-efficiency

(c) $\mathcal{I}$-efficiency (d) $\mathcal{I}_d$-efficiency

Figure 3: Efficiencies of the designs generated for scenario Block-1-MI.

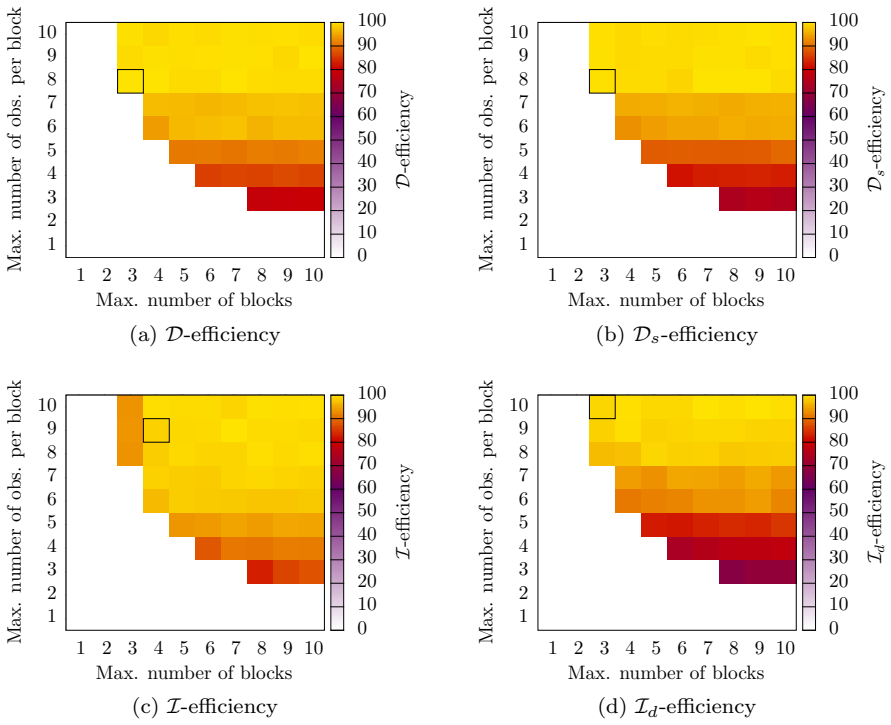


(a) $\mathcal{D}$-efficiency (b) $\mathcal{D}_s$-efficiency

(c) $\mathcal{I}$-efficiency (d) $\mathcal{I}_d$-efficiency

Figure 4: Efficiencies of the designs generated for scenario Block-2-MI.

The heat maps in Figure 4 are very similar to those shown in Figure 3. Designs involving a few large blocks have larger efficiencies than those involving many small blocks. For scenario Block-2-MI, the designs with the best $\mathcal{D}$-, $\mathcal{D}_s$- and $\mathcal{I}_d$-optimality criterion values involve 3 blocks of size 8. The design with the best $\mathcal{I}$-optimality criterion value, however, involves 6 blocks with a maximum size of 6. This design is 11.51% more $\mathcal{I}$-efficient than the design involving 3 blocks of size 8.



(a) $\mathcal{D}$-efficiency (b) $\mathcal{D}_s$-efficiency

(c) $\mathcal{I}$-efficiency (d) $\mathcal{I}_d$-efficiency

Figure 5: Efficiencies of the designs generated for scenario Block-2-MIQ.

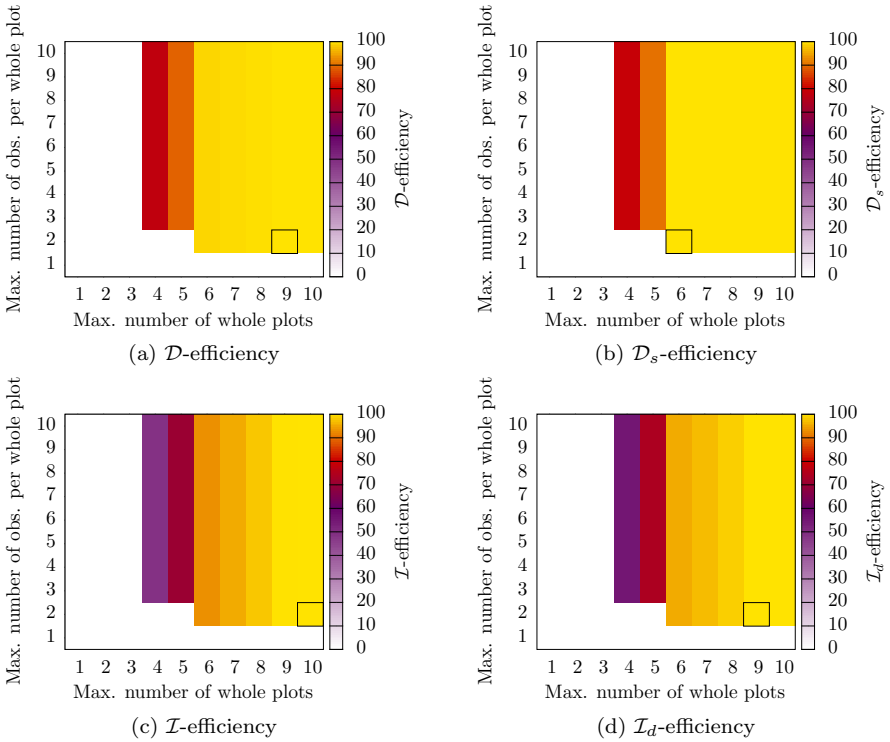
The heat maps in Figure 5 exhibit the same tendency: the best blocking configurations involve a few relatively large blocks. For scenario Block-2-MIQ, the designs with the best $\mathcal{D}$-, $\mathcal{D}_s$- and $\mathcal{I}_d$-optimality criterion values also involve 3 blocks. The design with the best $\mathcal{I}$-optimality criterion value, however, involves 4 blocks. Compared to scenario Block-2-MI, imposing constraints on the size of the blocks in scenario Block-2-MIQ has a stronger impact on the efficiencies of the designs. For instance, for scenario Block-2-MIQ, a design involving 6 blocks of size 4 is only 87.64% $\mathcal{I}$-efficient relative to the $\mathcal{I}$-optimal design with the best blocking configuration. This $\mathcal{I}$-efficiency is smaller than its counterpart for scenario Block-2-MI, which is 97.37%.

6.2 Optimal designs for split-plot experiments

6.2.1 Main-effects models

Figures 6 and 7 show the heat maps for scenarios Split-1-M and Split-2-M, respectively. These scenarios correspond to split-plot experiments for estimating main-effects models and involving a total number of observations equal to 12. These figures show that optimal designs for split-plot experiments (for estimating main-effects models) involve a relatively large number of whole plots of a relatively small size.

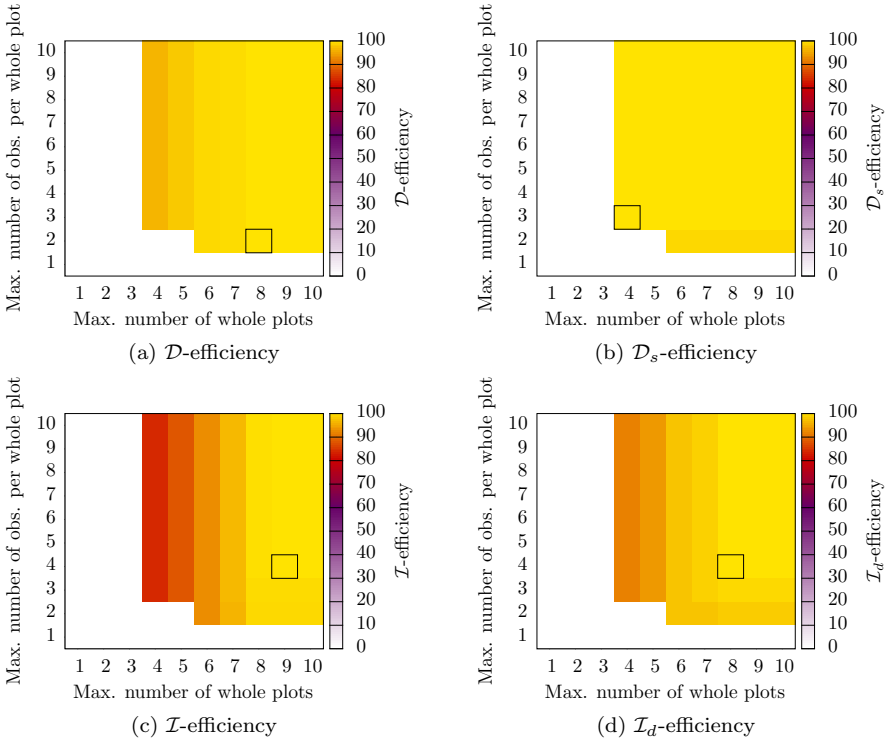
Figures 6(a), 6(c) and 6(d) show that, for scenario Split-1-M, the designs with the best $\mathcal{D}$-, $\mathcal{I}$- and $\mathcal{I}_d$-optimality criterion values involve 9 or 10 whole plots, some of which are of size 1. The design with the best



(a) $\mathcal{D}$-efficiency (b) $\mathcal{D}_s$-efficiency

(c) $\mathcal{I}$-efficiency (d) $\mathcal{I}_d$-efficiency

Figure 6: Efficiencies of the designs generated for scenario Split-1-M.



(a) $\mathcal{D}$-efficiency (b) $\mathcal{D}_s$-efficiency

(c) $\mathcal{I}$-efficiency (d) $\mathcal{I}_d$-efficiency

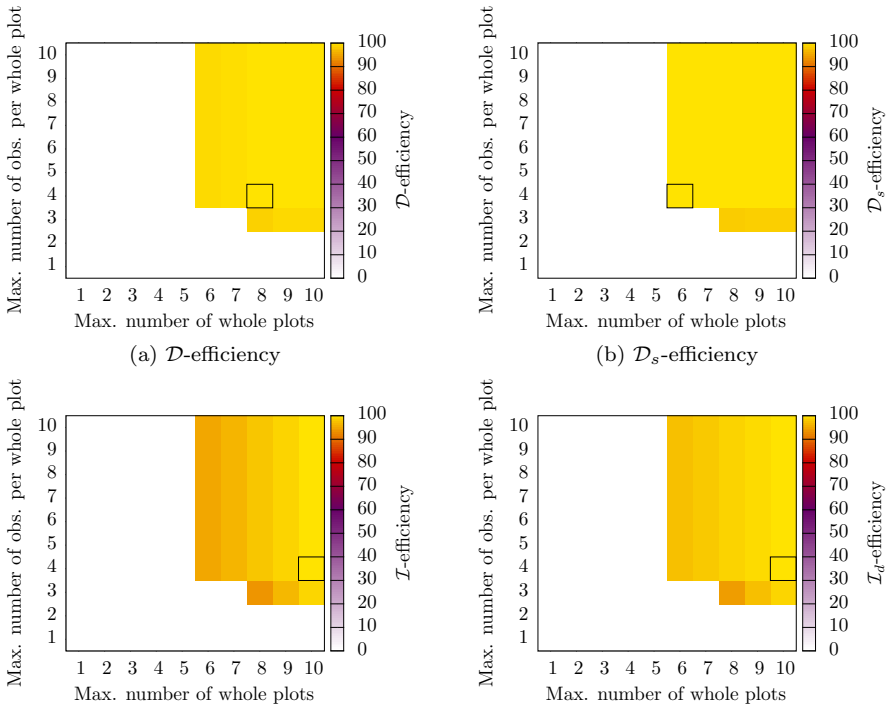
Figure 7: Efficiencies of the designs generated for scenario Split-2-M.

$\mathcal{D}_s$-optimality criterion value, however, involves 6 whole plots of size 2. Imposing constraints on the number of whole plots has a stronger impact on the $\mathcal{I}$- and the $\mathcal{I}_d$-efficiency of the designs, than on their $\mathcal{D}$- and $\mathcal{D}_s$-efficiency. For instance, a design involving 4 whole plots of size 3 is only 48.39% $\mathcal{I}$-efficient compared to the $\mathcal{I}$-optimal design with the best whole-plot configuration. In contrast, the same kind of design is 77.38% $\mathcal{D}$-efficient relative to the $\mathcal{D}$-optimal design with the best whole-plot configuration.

The heat maps in Figure 7 show similar patterns to those in Figure 6. Imposing small numbers of whole plots also has a stronger impact on the $\mathcal{I}$- and the $\mathcal{I}_d$-efficiency of the designs that on the $\mathcal{D}$- or the $\mathcal{D}_s$-efficiency. For scenario Split-2-M, the designs with the best $\mathcal{D}$-, $\mathcal{I}$- and $\mathcal{I}_d$-optimality values involve 8 or 9 whole plots. In fact, both the $\mathcal{I}$- and the $\mathcal{I}_d$-optimal designs have very particular whole-plot configurations. The former involves 8 whole plots of size 1 and 1 whole plot of size 4. The latter involves 6 whole plots of size 1, 1 whole plot of size 2 and 1 whole plot of size 4. The presence of a whole plot of size 4 is due to the 4-level categorical factor in scenario Split-2-M. In contrast, the design with the best $\mathcal{D}_s$-optimality criterion value involves 4 whole plots of size 3. Compared to scenario Split-1-M, imposing constraints on the number of whole plots has a milder effect on the efficiencies of the designs in scenario Split-2-M. For instance, for scenario Split-2-M, a design involving 4 whole plots of size 3 is 95.97% $\mathcal{D}$-efficient relative to the $\mathcal{D}$-optimal design with the best whole-plot configuration. A design for scenario Split-1-M with the same whole-plot structure is only 77.38% $\mathcal{D}$-efficient relative to the corresponding $\mathcal{D}$-optimal design.

6.2.2 Models with interaction effects and quadratic effects

Figures 8, 9 and 10 show the heat maps for scenarios Split-1-MI, Split-2-MI and Split-2-MIQ, respectively. Scenarios Split-1-MI and Split-2-MI correspond to split-plot experiments for models with main effects and interaction effects, and involve a total number of observations equal to 24. Scenario Split-2-MIQ is similar to scenario Split-2-MI, but the model also involves quadratic effects. The color patterns shown by the heat

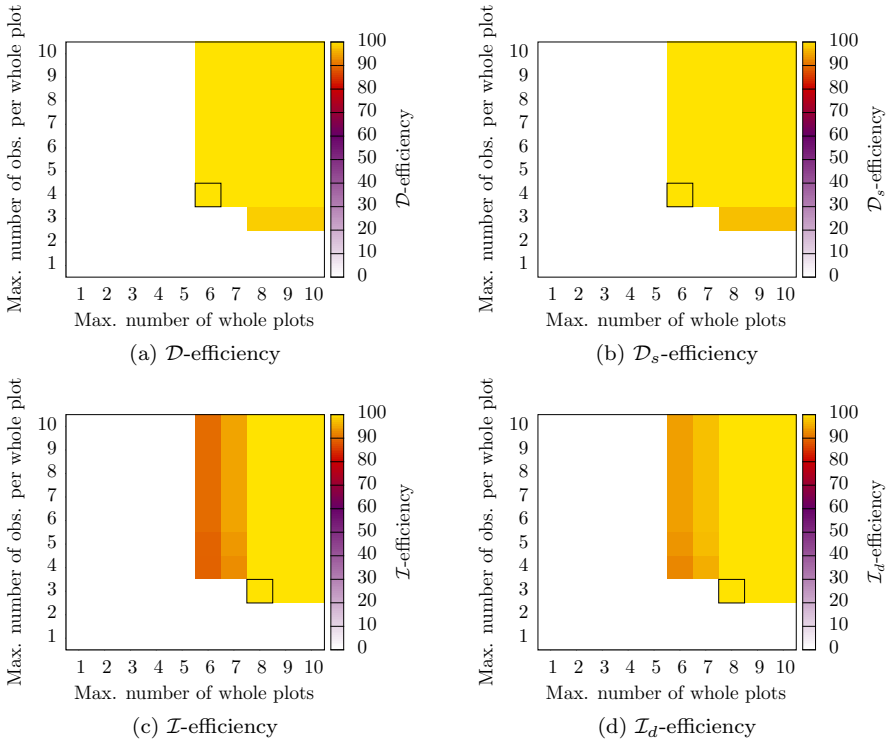


(a) $\mathcal{D}$-efficiency (b) $\mathcal{D}_s$-efficiency

(c) $\mathcal{I}$-efficiency (d) $\mathcal{I}_d$-efficiency

Figure 8: Efficiencies of the designs generated for scenario Split-1-MI.

maps in this section are quite uniform. This is different from the heat maps in Section 6.1.2 for blocked experiments involving the same kinds of models. This suggests that, when considering experiments involving these models, the generation of optimal split-plot designs is less challenging than the generation of optimal blocked designs. This is because the levels of the hard-to-change factors in split-plot designs must be constant for all observations in the same whole plot. For a given experimental scenario, the number of possible split-plot designs that can be generated is much smaller than the number of possible blocked designs. As a consequence, for split-plot scenarios, the algorithmic framework is required to explore a considerably smaller solution space, and it is able to find the optimal whole-plot configuration for all the combinations of both upper bounds.



(a) $\mathcal{D}$-efficiency (b) $\mathcal{D}_s$-efficiency

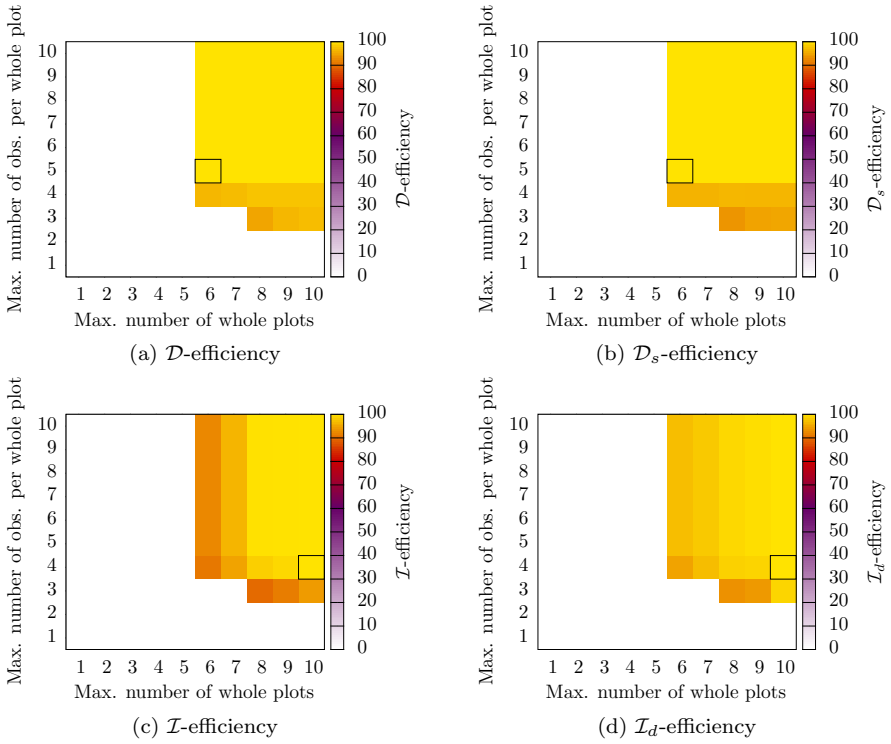
(c) $\mathcal{I}$-efficiency (d) $\mathcal{I}_d$-efficiency

Figure 9: Efficiencies of the designs generated for scenario Split-2-MI.

The heat maps in Figures 8 and 9 show that, for split-plot experiments for estimating interaction effects, imposing constraints on the number of whole plots has a low impact on the $\mathcal{D}$- and the $\mathcal{D}_s$-efficiency of the designs. For instance, for scenario Split-1-MI, the design with 6 whole plots of size 4 is 99.25% $\mathcal{D}$-efficient relative to the $\mathcal{D}$-optimal design with the best whole-plot configuration (involving 8 whole plots). For scenario Split-2-MI, this whole-plot configuration even leads to the design with the best $\mathcal{D}$-optimality criterion value. In contrast, as also observed in Section 6.2.1, imposing constraints on the number of whole plots has a stronger impact on the $\mathcal{I}$- and the $\mathcal{I}_d$-efficiency of the designs. For example, for scenario Split-2-MI, the design with 6 whole plots of size 4 is only 88.64% $\mathcal{I}$-efficient compared to the $\mathcal{I}$-optimal design with the best whole-plot configuration.

The heat maps in Figure 10 show the same tendency: imposing constraints on the number of whole plots has a considerable impact on the $\mathcal{I}$- and the $\mathcal{I}_d$-efficiency of the designs. Moreover, compared to scenario Split-2-MI, imposing constraints on the size of the whole plots has a greater impact on the efficiencies of the designs for scenario Split-2-MIQ. As a consequence, designs with whole plots of equal size tend to be of lower quality than designs with heterogeneous whole plots. For instance, the design involving 6 whole plots of size

4 is 96.10% $\mathcal{D}$-efficient relative to the $\mathcal{D}$-optimal design with the best whole-plot configuration (involving 2 whole plots of size 2 and 4 whole plots of size 5). The design with 6 whole plots of size 4 is only 90.54% $\mathcal{I}$-efficient relative to the $\mathcal{I}$-optimal design (involving 3 whole plots of size 1, 3 whole plots of size 2, 1 whole plot of size 3 and 3 whole plots of size 4).



(a) $\mathcal{D}$-efficiency (b) $\mathcal{D}_s$-efficiency

(c) $\mathcal{I}$-efficiency (d) $\mathcal{I}_d$-efficiency

Figure 10: Efficiencies of the designs generated for scenario Split-2-MIQ.

6.3 Final discussion

The results described in the previous sections confirm that the best grouping configuration for an optimal design strongly depends on the characteristics of the experiment. Moreover, the impact of tightening or relaxing the upper bounds on the number of groups and the number of observations per group varies from scenario to scenario. These are strong arguments in favour of using algorithmic techniques that are able to identify not only the best factor-level configurations, but also the best grouping configuration. Additionally, the results in this paper should also encourage experimenters to consider designs with different grouping configurations, whenever possible. The heat maps used to present our results have proven to be a very helpful tool that allows to perform a quick visual design comparison in these cases.

Despite the differences in the results obtained for each experimental scenario, there are some general tendencies in the computational simulations. First, for blocked experiments, imposing small numbers of blocks has a stronger impact on the $\mathcal{I}$-efficiency than on the $\mathcal{D}$-, $\mathcal{D}_s$- and $\mathcal{I}_d$-efficiency of the designs. This is due to the fact that the $\mathcal{I}$-optimality criterion pays much attention to estimating the intercept of the model. In most cases, imposing small numbers of blocks reduces the ability of a design to estimate this parameter precisely and therefore has a negative impact on the design's $\mathcal{I}$-efficiency. Second, for split-plot experiments, imposing small numbers of whole plots seems to have a stronger impact on the $\mathcal{I}$- and $\mathcal{I}_d$-efficiency than on the $\mathcal{D}$- and $\mathcal{D}_s$-efficiency of the designs. This phenomenon is in line with the observations made by Trinca and Gilmour (2015). It appears that the quality of the model's predictions (whether of the response or differences

in the response) strongly depends on the model parameters corresponding to the hard-to-change factors. For that reason, obtaining precise estimates of the parameters related to the hard-to-change factors seems to be of utmost importance. Achieving a good estimation of these parameters requires the hard-to-change factors to be varied more often, and therefore necessitates a larger number of whole plots in the designs. In contrast, in the calculation of the $\mathcal{D}$- and $\mathcal{D}_s$-optimality criteria, a poor estimation of the parameters related to the hard-to-change factors is compensated by a precise estimation of the parameters related to the easy-to-change factors.

7 Comparison to other algorithmic techniques

In this section, we compare the designs generated by our framework to those generated by other algorithms. First, we focus on different designs generated by the point-exchange algorithm of Goos and Vandebroek (2004) for a split-plot experiment involving 2 continuous factors. We then focus on optimal designs generated by the coordinate-exchange algorithm implemented in the statistical package JMP for the split-plot experiments described in Table 6.

7.1 Point-exchange algorithm

Goos and Vandebroek (2004) show different $\mathcal{D}$-optimal designs generated for a split-plot experiment involving 2 continuous factors (one hard-to-change factor and one easy-to-change factor), 10 observations and a full-quadratic model. These designs were generated using their point-exchange algorithm considering several experimental scenarios (i.e., different variance ratios and different restrictions on the number of whole plots). For all experimental scenarios, our framework generates the same designs as those generated by the point-exchange algorithm.

7.2 Statistical package JMP

The custom designer in JMP is able to generate $\mathcal{D}$- and $\mathcal{I}$-optimal designs for split-plot experiments by allowing the user to specify the number of whole plots in the design. We used this tool to generate optimal designs for the split-plot experiments described in Table 6. For each experiment, we generated several designs, one for each feasible number of whole plots in the range $\{1, \ldots, 10\}$. In order to do so, we configured the JMP software to execute 10000 random starts. From the resulting set of designs, we identified the one with the best optimality criterion value. In other words, we selected the design with the best whole-plot configuration with at most 10 whole plots. We then compared this design to the best one generated by our framework during the computational simulations (using 2000 random restarts) described in Section 6. Table 7 shows the statistical efficiencies of the optimal designs generated by JMP relative to those generated by our framework. The first column of the table shows the experimental scenario, the second column shows the efficiencies of the $\mathcal{D}$-optimal designs and the third column shows efficiencies of the $\mathcal{I}$-optimal designs. In half of the cases, the designs generated by our framework are statistically more efficient than those generated by JMP. In particular, the $\mathcal{I}$-optimal design for the experiment Split-2-MIQ is around 8% more efficient than that generated by JMP. This difference is due to the fact that JMP tends to generate designs in which the observations are distributed evenly across the whole plots. However, the $\mathcal{I}$-optimal whole-plot configuration involves whole plots with very different sizes: 3 whole plots of size 1, 3 whole plots of size 2, 1 whole plot of size 3 and 3 whole plots of size 4.

8 Comparison to completely randomized designs

In this section, we compare the statistical efficiency of optimal designs for two-stratum experiments to that of optimal designs for completely randomized experiments. To this end, we focus on the set of experiments

described in Table 6. We considered the $\mathcal{D}$-, $\mathcal{D}_s$-, $\mathcal{I}$- and $\mathcal{I}_d$-optimal designs with the best grouping configurations generated during the computational simulation described in Section 6. For each of the ten scenarios, we also generated completely randomized optimal designs (i.e., with as many groups as observations). The statistical efficiencies of the two-stratum designs relative to those of the completely randomized designs are shown in Table 8. Table 8(a) shows the efficiencies of the blocked designs, while Table 8(b) shows the efficiencies of the split-plot designs. In each table, the first column shows the experimental scenario, while the next columns show the relative $\mathcal{D}$-, $\mathcal{D}_s$-, $\mathcal{I}$- and $\mathcal{I}_d$-efficiencies.

Table 7: Efficiencies of the designs generated by JMP relative to the designs generated by our algorithmic framework.

Identifier	$\mathcal{D}$-opt.	$\mathcal{I}$-opt.
Split-1-M	100.00	100.00
Split-1-MI	98.44	97.29
Split-2-M	100.00	97.89
Split-2-MI	100.00	100.00
Split-2-MIQ	96.18	92.60
Average	98.92	97.55

Table 8: Efficiencies of optimal two-stratum designs relative to optimal completely randomized designs.

(a) Blocked experiments

Identifier	$\mathcal{D}$-opt.	$\mathcal{D}_s$-opt.	$\mathcal{I}$-opt.	$\mathcal{I}_d$-opt.
Block-1-M	159.84	200.00	147.56	200.00
Block-1-MI	173.34	194.09	152.69	194.96
Block-2-M	156.61	200.00	135.15	198.76
Block-2-MI	175.81	197.23	140.59	200.00
Block-2-MIQ	174.56	194.47	143.52	194.55
Average	168.03	197.15	143.90	197.65

(b) Split-plot experiments

Identifier	$\mathcal{D}$-opt.	$\mathcal{D}_s$-opt.	$\mathcal{I}$-opt.	$\mathcal{I}_d$-opt.
Split-1-M	103.64	110.09	100.38	104.42
Split-1-MI	124.33	130.94	109.55	114.32
Split-2-M	111.72	121.75	110.86	117.41
Split-2-MI	137.26	146.99	112.37	125.21
Split-2-MIQ	128.04	135.90	113.42	117.49
Average	121.00	129.13	109.32	115.77

Table 8 shows that two-stratum designs are considerably more efficient than completely randomized designs. These results are in line with those discussed by Goos and Vandebroek (2001a, 2004). Except in scenario Split-1-M, all two-stratum designs are at least 10% more efficient than their completely randomized counterparts. On average, blocked designs have larger statistical efficiencies than split-plot designs. This makes sense because split-plot designs can be viewed as blocked designs subject to additional constraints.

Similarly, designs for models involving interaction effects and quadratic effects have larger relative efficiencies than designs for main-effects models. This is mainly due to the grouping configurations of the optimal two-stratum designs in case interactions and quadratic effects are of interest: for these models, the optimal designs involve small numbers of groups. It is important to mention that these efficiencies are calculated considering a variance ratio $\sigma_\delta^2/\sigma_\varepsilon^2$ equal to one. As noted by Goos (2002) and Goos and Vandebroek (2004), the efficiencies relative to completely randomized designs are very sensitive to the value of this parameter (i.e., they strongly increase with $\sigma_\delta^2/\sigma_\varepsilon^2$).

9 Conclusions and recommendations

Two-stratum experiments are widely used in case a complete randomization is not possible. In some experimental scenarios, there are constraints that limit the number of observations that can be made under homogeneous conditions. In other scenarios, there are factors that are hard or expensive to change. In these cases, it is better to carry out an experiment that groups the observations. The most common strategy for the design of these kinds of experiments is to consider groups of equal size. The number of groups and the number of observations per group are usually defined by the constraints that limit the experimental scenario. However, we argue that these constraints do not define the design itself, but should be considered only as upper bounds. Limiting the experimental scenario only by upper bounds makes the task of generating an optimal design more complex and demanding. However, despite the extra complexity, the results in this paper show that the additional flexibility in the design generation process can significantly improve the quality of the designs. The results also show that the best grouping configuration for an optimal design depends on the characteristics of the two-stratum experiment (namely, the type of experiment, the model to be estimated and the optimality criterion considered). For that reason, experimenters require algorithmic generation techniques that are able to identify not only the best factor-level configurations, but also the best grouping configuration.

In this paper, we therefore propose an algorithmic framework to generate optimal designs for two-stratum experiments in which the number of groups and the number of observations per group are limited only by upper bounds. One main advantage is that it implements a coordinate-based approach and therefore does not require a set of candidate points to be specified. The execution of the algorithmic framework resembles that of a multi-level local search approach. The low level implements a modified version of the coordinate-exchange algorithm proposed by Jones and Goos (2007) and optimizes the factor-level configurations. The high level implements an algorithm similar to a variable neighborhood search and optimizes the grouping configuration. The core component of this high level is a flexible resizing operator. The structural modifications performed by this operator allow the algorithm to evaluate a wide range of grouping configurations. The results of our computational simulations show the benefits of this thorough exploration in terms of the quality of the designs generated, especially when the best grouping configuration for the design involves relatively few large groups.

Our computational simulations show that the impact of imposing constraints on the number of groups depends on the optimality criterion used to measure the quality of the design. Moreover, given an experimental scenario, considering different optimality criteria might lead to designs with different grouping configurations. This phenomenon introduces a new challenge, namely the generation of a design with the grouping configuration that offers the best trade-off between the optimality criteria of interest. In order to generate such a design, it would be useful to extend the algorithmic framework to perform a multi-objective optimization similar to that described by Sambo et al. (2014).

Appendix: Performance evaluation of the algorithmic framework considering different degrees of exploration

The high-level layer of the algorithmic framework (presented in Section 5.2) optimizes the grouping configuration of the designs by applying a resizing operator. The number of times T the operator is applied to a given pair of groups determines how thorough the optimization process is. Initially, this layer uses $T = 1$. However, the value of T is increased when the algorithm gets stuck in a locally optimal design and no further improvements can be made. This allows the algorithm to perform a more thorough exploration of the solution space in order to escape from the local optimum. The T value is progressively increased until a user-specified maximum value is reached. This maximum value therefore defines the degree of exploration performed by the algorithmic framework.

In this appendix, we show the results of a computational simulation carried out to study the performance of the algorithmic framework for different T values. The simulation was executed on a computer with a 2.93 GHz Intel Core i7 processor. The algorithm was executed using 2000 restarts for each of the experimental scenarios described in Table 6. We fixed the maximum number of groups and the maximum group size to 10, and we considered the $\mathcal{D}$- and the $\mathcal{I}$-optimality criteria. We evaluated the performance of the framework considering two different measures: the average statistical efficiency of the designs generated in each restart (relative to the optimal design), and the average execution time.

The results of the performance evaluation are shown in Tables 9 and 10. The first column of each table identifies the experimental scenario. The next columns show the values of the performance measures for each of the maximum values of T considered. The label $\max(N_i)$ shown in the last column header represents a value that is equal to the size of the largest group in the design. This value causes the algorithm to perform the most exhaustive exploration possible.

Table 9: Effects of the maximum value of T on the performance of the algorithmic framework in terms of $\mathcal{D}$-efficiency.

	1		2		3		$\max(N_i)$	
Identifier	$\mathcal{D}$-eff.	Time (s)	$\mathcal{D}$-eff.	Time (s)	$\mathcal{D}$-eff.	Time (s)	$\mathcal{D}$-eff.	Time (s)
Block-1-M	89.07	0.01	89.34	0.01	89.20	0.01	89.38	0.01
Block-1-MI	87.90	0.44	88.53	0.67	88.74	0.83	89.37	1.24
Block-2-M	89.97	0.01	90.28	0.02	90.26	0.03	90.45	0.03
Block-2-MI	89.40	0.51	89.86	0.78	90.18	0.98	90.65	1.63
Block-2-MIQ	84.81	0.70	85.63	0.82	85.91	1.01	87.28	2.00
Split-1-M	97.96	0.02	97.93	0.02	98.01	0.02	98.00	0.02
Split-1-MI	98.59	0.91	98.67	1.13	98.68	1.21	98.68	1.23
Split-2-M	97.63	0.02	97.64	0.03	97.63	0.03	97.64	0.03
Split-2-MI	97.02	0.73	97.22	0.94	97.26	1.22	97.28	1.24
Split-2-MIQ	96.00	0.99	96.35	1.37	96.43	1.56	96.48	1.75

For split-plot experiments, the algorithmic framework generates, on average, designs with higher quality than for the blocked experiments. For instance, the average $\mathcal{D}$-efficiency of the designs generated for the experimental scenario Block-1-M is around 90%, while, for scenario Split-1-M, it is around 98%. This again indicates that the generation of optimal designs for blocked experiments is harder, which is in line with the results discussed in Section 6.1.1. Additionally, the generation of $\mathcal{I}$-optimal designs requires longer computing times than the generation of $\mathcal{D}$-optimal designs. This is due to the inversion of the information matrix

$f'(\boldsymbol{X})\boldsymbol{V}^{-1}f(\boldsymbol{X})$ required by the $\mathcal{I}$-optimality criterion.

Table 10: Effects of the maximum value of T on the performance of the algorithmic framework in terms of $\mathcal{I}$-efficiency.

	1		2		3		$\max(N_i)$	
Identifier	$\mathcal{I}$-eff.	Time (s)	$\mathcal{I}$-eff.	Time (s)	$\mathcal{I}$-eff.	Time (s)	$\mathcal{I}$-eff.	Time (s)
Block-1-M	86.45	0.01	87.06	0.01	87.40	0.01	88.00	0.02
Block-1-MI	86.46	0.66	87.58	1.01	88.11	1.30	89.05	1.75
Block-2-M	92.68	0.02	93.14	0.03	93.47	0.03	94.13	0.04
Block-2-MI	87.88	0.98	89.02	1.58	89.62	1.99	90.52	2.85
Block-2-MIQ	85.72	1.22	86.83	1.68	87.35	2.02	88.71	2.80
Split-1-M	96.33	0.02	96.34	0.03	96.35	0.03	96.40	0.03
Split-1-MI	96.91	1.45	97.10	1,94	97.09	2.12	97.10	2.15
Split-2-M	94.97	0.03	95.07	0.04	95.06	0.04	95.09	0.05
Split-2-MI	93.33	1.54	93.59	1.92	93.71	2.11	93.80	2.15
Split-2-MIQ	91.52	1.68	92.04	2.28	92.20	2.56	92.33	2.78

The maximum value of T has a strong impact on the execution time of the algorithm. It is reasonable to expect that the more thorough the exploration, the longer the execution time. The increase in execution time is more apparent in the generation of designs that involve groups of relatively large size (i.e., designs for experimental scenarios involving models with interaction effects and quadratic effects). The optimal designs for experiments involving main-effects models involve smaller groups, and, for their generation, the algorithm never applies the resizing operator more than once or twice. For this reason, the maximum value of T used by the algorithm is essentially a small number, and the computing time does not increase by changing the maximum value for T. The impact on the average statistical efficiency of the designs generated follows a similar pattern: the increase in $\mathcal{D}$- and $\mathcal{I}$-efficiency is more pronounced for the generation of designs involving groups of large size, i.e., for blocked experiments and models with interaction effects and quadratic effects. These designs are composed of 3 or 4 groups only, and have around 6 observations per group. For that reason, considering larger values of T allows a better exploration of the design space, and leads to designs with better quality. For instance, by considering only $T = 1$, the average $\mathcal{I}$-efficiency of the designs generated for scenario Block-1-MI is 86.46%. The larger the maximum value of T, the better the quality of the designs generated. By allowing the framework to perform the most exhaustive exploration, the average $\mathcal{I}$-efficiency increases to 89.05%. In conclusion, the larger the size of the groups that compose the optimal design, the larger the impact of allowing the algorithm to perform a more thorough exploration, both in terms of the quality of the designs generated and the execution time of the algorithm. Since the best grouping configuration of an optimal design is not known in advance, we recommend specifying a large maximum value of T and allow the framework to perform a thorough exploration.

Acknowledgements

This research is supported by the Interuniversity Attraction Poles (IAP) Programme initiated by the Belgian Science Policy Office (COMEX project).

References

Frank T. Anbari and James M. Lucas. Super-efficient designs: how to run your experiment for higher efficiency and lower cost. In *Annual Quality Congress Proceedings-American Society for Quality Control*, pages 853–853, 1994.

Frank T. Anbari and James M. Lucas. Designing and running super-efficient experiments: Optimum blocking with one hard-to-change factor. *Journal of Quality Technology*, 40(1):31–45, 2008.

Virgil L. Anderson and Robert A. McLean. *Design of experiments: a realistic approach*, volume 5. CRC Press, 1974.

Heidi Arnouts and Peter Goos. Staggered-level designs for experiments with more than one hard-to-change factor. *Technometrics*, 54(4):355–366, 2012.

Heidi Arnouts and Peter Goos. Staggered-level designs for response surface modeling. *Journal of Quality Technology*, 47(2):156–175, 2015.

Heidi Arnouts, Peter Goos, and Bradley Jones. Design and analysis of industrial strip-plot experiments. *Quality and Reliability Engineering International*, 26(2):127–136, 2010.

Anthony C. Atkinson and Alexander N. Donev. The construction of exact D-optimum experimental designs with application to blocking response surface designs. *Biometrika*, 76(3):515–526, 1989.

Anthony C. Atkinson, Alexander N. Donev, and Randall Tobias. *Optimum experimental designs, with SAS*, volume 34. Oxford University Press, USA, 2007.

Derek Bingham and Randy R. Sitter. Minimum-aberration two-level fractional factorial split-plot designs. *Technometrics*, 41(1):62–70, 1999.

Derek Bingham, Eric D. Schoen, and Randy R. Sitter. Designing fractional factorial split-plot experiments with few whole-plot factors. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 53(2): 325–339, 2004.

Derek Bingham, Eric D. Schoen, and Randy R. Sitter. Corrigendum: Designing fractional factorial split-plot experiments with few whole-plot factors. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 54(5):955–958, 2005.

Soren Bisgaard. Blocking generators for small 2^{k-p} designs. *Journal of Quality Technology*, 26(4):288–296, 1994.

Neil A. Butler. Construction of two-level split-plot fractional factorial designs for multistage processes. *Technometrics*, 46(4):445–451, 2004.

Scott D. Chasalow. *Exact response surface designs with random block effects*. PhD thesis, University of California, Berkeley, 1992.

Shao Wei Cheng and C. F. Jeff Wu. Choice of optimal blocking schemes in two-level and three-level designs. *Technometrics*, 44(3), 2002.

R. Dennis Cook and Christopher J. Nachtsheim. Computer-aided blocking of factorial and response-surface designs. *Technometrics*, 31(3):339–346, 1989.

Valerii V. Fedorov. *Theory of optimal experiments*. Academic Press, New York, 1972.

Steven G. Gilmour and Peter Goos. Analysis of data from non-orthogonal multistratum designs in industrial experiments. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 58(4):467–484, 2009.

Peter Goos. *The optimal design of blocked and split-plot experiments*, volume 164. Springer, New York, 2002.

Peter Goos. Optimal versus orthogonal and equivalent-estimation design of blocked and split-plot experiments. *Statistica Neerlandica*, 60(3):361–378, 2006.

Peter Goos and Bradley Jones. *Optimal design of experiments: A case study approach*. Wiley, 2011.

Peter Goos and Martina Vandebroek. Optimal split-plot designs. *Journal of Quality Technology*, 33(4): 436–450, 2001a.

Peter Goos and Martina Vandebroek. D-optimal response surface designs in the presence of random block effects. *Computational Statistics & Data Analysis*, 37(4):433–453, 2001b.

Peter Goos and Martina Vandebroek. D-optimal split-plot designs with given numbers and sizes of whole plots. *Technometrics*, 45(3):235–245, 2003.

Peter Goos and Martina Vandebroek. Outperforming completely randomized designs. *Journal of Quality Technology*, 36(1):12–26, 2004.

Pierre Hansen and Nenad Mladenović. Variable neighborhood search. In Fred Glover and Gary A. Kochenberger, editors, *Handbook of Metaheuristics*, volume 57 of *International Series in Operations Research & Management Science*, pages 145–184. Springer, 2003.

Peng Huang, Dechang Chen, and Joseph O. Voelkel. Minimum-aberration two-level split-plot designs. *Technometrics*, 40(4):314–326, 1998.

Bradley Jones and Peter Goos. A candidate-set-free algorithm for generating D-optimal split-plot designs. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 56(3):347–364, 2007.

Bradley Jones and Peter Goos. D-optimal design of split-split-plot experiments. *Biometrika*, 96(1):67–82, 2009.

Bradley Jones and Christopher J. Nachtsheim. Split-plot designs: What, why, and how. *Journal of Quality Technology*, 41(4):340–361, 2009.

Roselinde Kessels, Peter Goos, and Martina Vandebroek. Optimal designs for conjoint experiments. *Computational Statistics & Data Analysis*, 52(5):2369–2387, 2008.

Andre I. Khuri. Response surface models with random block effects. *Technometrics*, 34(1):26–37, 1992.

Jennifer D. Letsinger, Raymond H. Myers, and Marvin Lentner. Response surface methods for bi-randomization structures. *Journal of Quality Technology*, 28(4):381–397, 1996.

Robert W. Mee and Rodney L. Bates. Split-plot designs: Experiments for multistage batch processes. *Technometrics*, 40(2):127–140, 1998.

Ruth K. Meyer and Christopher J. Nachtsheim. The coordinate-exchange algorithm for constructing exact optimal experimental designs. *Technometrics*, 37(1):60–69, 1995.

Arden Miller. Strip-plot configurations of fractional factorials. *Technometrics*, 39(2):153–161, 1997.

Nenad Mladenović and Pierre Hansen. Variable neighborhood search. *Computers & Operations Research*, 24(11):1097–1100, 1997.

Kalliopi Mylona, Peter Goos, and Bradley Jones. Optimal design of blocked and split-plot experiments for fixed effects and variance component estimation. *Technometrics*, 56(2):132–144, 2014.

Daniel Palhazi Cuervo, Peter Goos, and Kenneth Sörensen. Optimal design of large-scale screening experiments: a critical look at the coordinate-exchange algorithm. *Statistics and Computing*, 26(1):15–28, 2016.

Francesco Sambo, Matteo Borrotti, and Kalliopi Mylona. A coordinate-exchange two-phase local search algorithm for the D- and I-optimal designs of split-plot experiments. *Computational Statistics & Data Analysis*, 71:1193–1207, 2014.

Randy R Sitter, Jiahua Chen, and Moshe Feder. Fractional resolution and minimum aberration in blocked 2^{n-k} designs. *Technometrics*, 39(4):382–390, 1997.

Byran J. Smucker, Enrique del Castillo, and James L. Rosenberger. Model-robust designs for split-plot experiments. *Computational Statistics & Data Analysis*, 56(12):4111–4121, 2012.

Don X. Sun, C. F. Jeff Wu, and Yuanhao Chen. Optimal blocking schemes for 2^n and 2^{n-p} designs. *Technometrics*, 39(3):298–307, 1997.

Luzia A Trinca and Steven G. Gilmour. Difference variance dispersion graphs for comparing response surface designs with applications in food technology. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 48(4):441–455, 1999.

Luzia A. Trinca and Steven G. Gilmour. An algorithm for arranging response surface designs in small blocks. *Computational Statistics & Data Analysis*, 33(1):25–43, 2000a.

Luzia A. Trinca and Steven G. Gilmour. An algorithm for arranging response surface designs in small blocks (erratum). *Computational Statistics & Data Analysis*, 33(1):25–43, 2000b.

Luzia A. Trinca and Steven G. Gilmour. Multistratum response surface designs. *Technometrics*, 43(1):25–33, 2001.

Luzia A. Trinca and Steven G. Gilmour. Improved split-plot and multi-stratum designs. *Technometrics*, 57 (2):145–154, 2015.

Runchu Zhang and DongKwon Park. Optimal blocking of two-level fractional factorial designs. *Journal of Statistical Planning and Inference*, 91(1):107–121, 2000.