DEPARTMENT OF ENGINEERING MANAGEMENT

An integrated algorithm for the optimal design of stated choice experiments with partial profiles

Daniel Palhazi Cuervo, Roselinde Kessels, 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 integrated algorithm for the optimal design of stated choice experiments with partial profiles

Daniel Palhazi Cuervo, Roselinde Kessels, Peter Goos & Kenneth Sörensen

RESEARCH PAPER 2015-004
JANUARY 2015

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/2015/1169/004

An integrated algorithm for the optimal design of stated choice experiments with partial profiles

Daniel Palhazi Cuervo[1], Roselinde Kessels[2], Peter Goos[1,3] and Kenneth Sörensen[1]

[1]University of Antwerp, Faculty of Applied Economics, Department of Engineering Management
[2]University of Antwerp, Faculty of Applied Economics, Department of Economics & StatUa Center for Statistics
[3]University of Leuven, Faculty of Bioscience Engineering & Leuven Statistics Research Center

January 26, 2015

Abstract

Stated choice experiments are conducted to identify the attributes that drive people's preferences when choosing between competing options of products or services. They are widely used in transportation in order to support the decision making of companies and governmental authorities. A large number of attributes might increase the complexity of the choice task in a choice experiment, and have a detrimental effect on the quality of the results obtained. In order to reduce the cognitive effort required by the experiment, researchers may resort to experimental designs where the levels of some attributes are held constant within a choice situation. These designs are called partial profile designs. In this paper, we propose an integrated algorithm for the generation of D-optimal designs for stated choice experiments with partial profiles. This algorithm optimizes the set of constant attributes and the levels of the varying attributes simultaneously. An extensive computational experiment shows that the designs produced by the integrated algorithm outperform those obtained by existing algorithms, and match the optimal designs that have been analytically derived for a number of benchmark instances. We also evaluate the performance of the algorithm under varying experimental conditions and study the structure of the designs generated.

Keywords: Stated choice experiments, Multinomial logit model, Partial profiles, (Bayesian) D-optimality, Utility-neutral designs, Coordinate-exchange algorithm.

1 Introduction

Stated choice (SC) experiments are widely used to study how people make choices and to identify the elements that drive people's preferences. They are performed in a wide range of applied fields in order to evaluate the trade-offs that people make when choosing between competing options. In a SC experiment, the product or service under study is characterized by a set of attributes, and each alternative, called *profile*, is described as a combination of attribute levels. Each respondent is presented with several groups of profiles, called *choice situations*, and asked to choose the profile of his/her preference in each situation. The main goal of a SC experiment is to estimate the importance of each attribute level from the repeated choices made. By doing so, it is possible to make predictions concerning a population's choice behaviour.

SC experiments are widely used in transportation in order to collect data for the study of travelling behaviour (see, for example, Tilahun et al. (2007), Hensher and Rose (2007) and Hensher (2008)) and for assisting governmental authorities in their policy-making activities (see, for example, Wang et al. (2002) and Saleh and Farrell (2005)). Among other things, SC experiments are carried out to study preferences for both air and ground transportation. For example, Ahern and Tapley (2008) studied the preferences of passengers for interurban routes in Ireland. They focused on identifying the opportunities of improvement and contrasting the services offered by trains and buses. The SC experiment considered five attributes: cost, journey length, frequency, reliability of the service and the presence of toilet facilities. Bliemer and Rose (2011) studied the preferences of airline customers. Their work has been motivated by the fact that flying into main airports is becoming more expensive, both for travellers and airlines. For that reason, there has been an increase in the use of regional airports that are located in the area surrounding major cities. Their main goal was to examine the trade-offs that travellers make between ticket price and overall travel time. The SC experiment involved six attributes: airline, ticket price, departure time, transfer time, egress price and egress time. Bliemer and Rose (2011) also summarized the literature using SC experiments published in the most important transportation journals during the period from January 2000 to August 2009. A total of 64 research papers are listed, resulting in 61 unique experimental designs. A general overview of the traditional design generation techniques, along with some algorithmic approaches, is given by Rose and Bliemer (2009).

The importance of each attribute is usually quantified using a multinomial logit (MNL) model or a generalization thereof that is built on the assumption that humans attempt to maximize the total utility when making a choice. In the MNL model, the total utility is expressed as a function of the utilities associated with each attribute, which are called *part-worth values* in the case of main-effects models. The MNL model implicitly assumes that humans are willing to make compensatory decisions. In simple terms, the positive utility due to preferred levels for some attributes compensates the negative utility due to undesired levels for other attributes. The use of non-compensatory decision rules (where, for example, respondents focus on one attribute and ignore the others) is in conflict with this assumption and renders the model's predictions questionable. One main reason why respondents may resort to non-compensatory strategies is the complexity of the decision task. Large numbers of attributes increase the required cognitive effort and might lead the respondents to use simpler decision strategies (Caussade et al., 2005; Hensher, 2006a,b). In order to reduce the complexity of the comparison and to prevent respondents from using simpler strategies, it is possible to hold the levels of some attributes constant in every choice situation. Since these profiles direct the attention only to the subset of attributes the levels of which are allowed to vary, they are called *partial profiles* (Chrzan, 2010). The number of varying attributes in a partial profile is called the *profile strength* (Großmann et al., 2006).

Several SC experiments in transportation involve large numbers of attributes. For example, an experiment carried out by Anderson et al. (2006) for the study of tourist parking preferences involved seven attributes. Loo et al. (2006) carried out an experiment to study the preferences of the public light bus industry regarding the introduction of alternative fuel vehicles in Hong Kong. This experiment involved eight attributes related to the vehicle attributes and seven attributes related to the government support. In other studies, the researchers have reduced or limited the number of attributes considered in their experiments in order to avoid an overwhelming complexity (see, for example, Hunt and Abraham (2007) and Sener et al. (2009)). The study of commute mode choice carried out by Bhat and Sardesai (2006) is an extreme example. During the pilot phase, respondents indicated that the experiment was too burdensome and recommended a reduction in the number of attributes and the number of choice situations. This suggestion encouraged the researchers to consider five attributes instead of the initial eight. More recently, SC experiments with partial profiles have been used in order to avoid the use of non-compensatory decision strategies due to large numbers of attributes. For example, Kupfer et al. (2013) studied the effect of six attributes on the airport choice of air freight service providers in Europe. In order to keep the choice tasks manageable for the respondents, they used partial profiles showing only four of the six attributes in each choice situation. Another example in which a partial profile design has proven useful, is the SC experiment used by Verhetsel et al. (2015). They quantified the impact of the different dimensions of accessibility on the location decision process of logistics companies. This study also involved six attributes, four of which were shown in each choice situation.

The MNL model is nonlinear in the parameters. This implies that the quality of a design for a SC experiment depends on the unknown parameters. Optimal designs can be constructed considering only one specific set of parameter values; these designs are referred to as *locally optimal designs* (Huber and Zwerina, 1996). Moreover, the nonlinear model can be treated as a linear one by assuming that all the parameter values are equal to zero. This assumption implies that the respondents have no preference for any of the attribute levels and all utilities are equal to zero. Hence, the respondents are equally likely to choose any of the alternatives. The designs produced under this assumption are called *utility-neutral (UN) designs.* A sound alternative to the locally optimal designs is the Bayesian approach, which accounts for the uncertainty about the true values of the parameters (Sándor and Wedel, 2001). This approach evaluates the quality of a design assuming a prior distribution for the parameter values. It has been shown that designs generated using a Bayesian approach generally outperform locally optimal and UN designs in terms of the quality of the parameter estimates (Sándor and Wedel, 2001; Kessels et al., 2011a,c, 2015). For this reason, Bayesian designs now constitute the state of the art for the generation of optimal designs for SC experiments (Bliemer and Rose, 2010).

In order to design a SC experiment with partial profiles, it is necessary to make two important decisions for each choice situation. First, the set of attributes the levels of which are held constant, has to be determined. Second, the levels of the varying attributes have to be selected. A substantial part of the literature related to the generation of SC experiments with partial profiles deals with UN designs: Graßhoff et al. (2004) generate optimal UN designs using orthogonal arrays for SC experiments where all attributes have the same number of levels and where there are two profiles per choice situation. Großmann et al. (2009) analytically derive UN designs for experiments with two profiles per choice situation and attributes with different numbers of levels. They focus on experiments with two groups of attributes where the number of attribute levels is fixed for every group. More recently, Großmann et al. (2014) extended their previous work to experiments with three groups of attributes. A problem with their approach is that the numbers of choice situations that would be required to implement these designs are prohibitively large.

The set of optimal designs that can be generated using the approach of Großmann et al. (2009) and Großmann et al. (2014) is very restricted. Only designs for experiments with two profiles per choice situation, specific combinations of attribute levels, and specific numbers of choice situations and constant attributes, can be generated. In response to these limitations, Kessels et al. (2011b) proposed a more flexible two-stage algorithm for the generation of Bayesian optimal designs. The algorithm is capable of generating designs with any number of choice situations, profiles per choice situation and constant attributes, and with attributes with any numbers of levels. During its first stage, the algorithm determines the set of constant attributes for each choice situation by generating a structure similar to that of a balanced incomplete block design. By doing so, the algorithm attempts to balance the number of times each attribute is held constant. In its second stage, the algorithm determines the levels of the varying attributes by applying a modified version of the coordinate-exchange algorithm (Meyer and Nachtsheim, 1995).

We believe that the approach of Kessels et al. (2011b) has three major weaknesses:

1. As pointed out by Großmann (2013), the two-stage algorithm is unable to match the quality of the optimal UN designs in the benchmark set of experiments proposed by Großmann et al. (2009). This is mainly because the selection of constant attributes of the designs generated by the algorithm is different from that of the optimal designs analytically generated.

2. The choice of the constant attributes and that of the levels of the varying attributes are made independently. More importantly, the choice of the constant attributes never changes during the execution of the algorithm. It is well known from the operations research literature that sequential algorithms (those that make each decision required for solving a problem in a different stage) are prone to generate suboptimal solutions. This is because the optimization performed in each stage is severely affected by the decisions made in previous stages (Salhi and Rand, 1989). Several empirical studies confirm that

integrated methods outperform sequential methods for the solution of a wide range of combinatorial optimization problems (Nagy and Salhi, 2007; Basten et al., 2012; Schittekat et al., 2013; Vidal et al., 2013).

3. When generating Bayesian optimal designs, the choice of the constant attributes is made independently from the prior information concerning the model parameters. For example, consider the case in which a SC experiment has two attributes with the same number of levels, but the prior variance of the parameters corresponding to the first attribute is much larger than the prior variance of the parameters corresponding to the second attribute. Due to the difference in uncertainty about the attributes' parameters, it is reasonable to expect the attributes to be held constant a different number of times. The two-stage algorithm ignores the prior distribution of parameter values when generating a Bayesian design. Therefore, designs accounting for differences in prior parameter variances in the selection of the constant attributes are not considered by that algorithm.

Kessels et al. (2015) improve the first phase of the two-stage algorithm by incorporating two new strategies for selecting the constant attributes. Instead of paying an equal amount of attention to each attribute, these two strategies pay an equal amount of attention to each attribute *level*. This results in statistically more efficient designs that vary the attributes with large numbers of levels more often than attributes with fewer levels. Nevertheless, the designs generated by the two-stage algorithm are still not able to match the quality of those analytically generated by Großmann et al. (2009).

In this paper, we propose an *integrated algorithm* for the generation of optimal designs for SC experiments with partial profiles. This algorithm overcomes the weaknesses of the algorithms of Kessels et al. (2011b) and Kessels et al. (2015). To this end, it optimizes the set of constant attributes and the levels of the varying attributes simultaneously. The integrated algorithm generates, in a short computing time, designs that outperform those produced by the algorithm of Kessels et al. (2015). However, for some benchmark instances, the integrated algorithm just misses the optimal design and produces a design that is slightly less efficient. To overcome this minor limitation, we also describe a more exhaustive algorithm that generates the optimal designs for nearly all cases. Furthermore, we discuss the results of an extensive computational experiment that was carried out with two main objectives: first, to asses the performance of the algorithms under different experimental conditions; and second, to analyse the effects that such conditions have on the structure of the designs generated.

This paper is organized as follows. Section 2 reviews the multinomial logit model and the optimality criteria considered to generate SC experiments with partial profiles. The original integrated algorithm is described in Section 3 and the exhaustive integrated algorithm is presented in Section 4. Section 5 explains the implementation details considered in order to speed up the execution of the algorithms. The computational experiments and the results obtained are discussed in Section 6. Section 7 revisits a SC experiment described in the transportation literature, and explains how its design can be improved by using the integrated algorithm. The final conclusions and some recommendations for future research are presented in Section 8.

2 The multinomial logit model and optimality criteria

The MNL model defines the utility U that a respondent gives to a profile $j, j = 1, \ldots, J$, in a choice situation $s, s = 1, \ldots, S$, as the sum of a systematic and a random component:

$$U_{s,j} = \boldsymbol{f}'(\boldsymbol{x}_{s,j})\boldsymbol{\beta} + \varepsilon_{s,j}. \tag{1}$$

The first term in this sum corresponds to the systematic component associated with the levels of the F attributes in the SC experiment: $\boldsymbol{x}_{s,j}$ is the $F \times 1$ vector of attribute levels for profile j in choice situation

s, $\boldsymbol{f}(\boldsymbol{x}_{s,j})$ is the $K \times 1$ vector containing the K effects-type coded attribute levels, and $\boldsymbol{\beta}$ is the $K \times 1$ vector of parameter values for the main effects or part-worths of the attribute levels. The vector $\boldsymbol{\beta}$ is the same for every respondent, which implies that respondents' preferences are assumed to be homogeneous. The second term in Equation (1) corresponds to the random component: $\varepsilon_{s,j}$ is an error term which is assumed to be independent and identically Gumbel distributed. The MNL probability that a respondent chooses profile j in choice situation s is then

$$p_{s,j} = \frac{\exp(\boldsymbol{f}'(\boldsymbol{x}_{s,j})\boldsymbol{\beta})}{\sum_{t=1}^{J}(\exp(\boldsymbol{f}'(\boldsymbol{x}_{s,t})\boldsymbol{\beta}))}, \tag{2}$$

where $\boldsymbol{\beta}$ is estimated using maximum likelihood.

The generation of an optimal design $\boldsymbol{X}$ for a SC experiment involves the generation of S submatrices $\boldsymbol{X}_s = [\boldsymbol{x}_{s,1}, \ldots, \boldsymbol{x}_{s,J}]'$ and depends on the Fisher information matrix

$$\boldsymbol{M}(\boldsymbol{X}, \boldsymbol{\beta}) = \sum_{s=1}^{S} \boldsymbol{f}'(\boldsymbol{X}_s)(\boldsymbol{P}_s - \boldsymbol{p}_s\boldsymbol{p}_s')\boldsymbol{f}(\boldsymbol{X}_s), \tag{3}$$

where $\boldsymbol{f}(\boldsymbol{X}_s) = [\boldsymbol{f}(\boldsymbol{x}_{s,1}), \ldots, \boldsymbol{f}(\boldsymbol{x}_{s,J})]'$, $\boldsymbol{p}_s = [p_{s,1}, \ldots, p_{s,J}]'$ and $\boldsymbol{P}_s = \text{diag}[p_{s,1}, \ldots, p_{s,J}]$. The information matrix summarizes how much information a design for a SC experiment contains concerning the respondents' preferences. By making an appropriate choice for the design, the information content of the SC experiment can be maximized. The most common approach to achieve this is to seek the design that maximizes the logarithm of the determinant of the information matrix. This approach is called the $\mathcal{D}$-optimal design approach. The problem with this approach is that the information matrix depends on the parameter values $\boldsymbol{\beta}$ through the probabilities $p_{s,j}$, and these values are unknown prior to the SC experiment. In order to deal with this dependency, it is possible to consider a single prior guess $\boldsymbol{\beta}_P$ of $\boldsymbol{\beta}$ and generate a so-called locally optimal design. These designs are called $\mathcal{D}_P$-optimal and maximize the logarithm of the determinant of the information matrix, $\mathcal{D}_P(\boldsymbol{X}) = \log|\boldsymbol{M}(\boldsymbol{X}, \boldsymbol{\beta}_P)|$.

Utility-neutral (UN) designs are special cases of locally optimal designs where $\boldsymbol{\beta}_P = \boldsymbol{0}_K$, with $\boldsymbol{0}_K$ a $K \times 1$ vector of zeroes. Using this prior vector is equivalent to assuming that the respondents have no preference for any of the profiles. This assumption causes the probabilities in Equation (2) to be equal to $1/J$ for all the profiles in a choice situation. The information matrix then reduces to

$$\boldsymbol{M}(\boldsymbol{X}) = J^{-1} \sum_{s=1}^{S} (\boldsymbol{f}'(\boldsymbol{X}_s)\boldsymbol{f}(\boldsymbol{X}_s) - J^{-1}(\boldsymbol{f}'(\boldsymbol{X}_s)\boldsymbol{1}_J)(\boldsymbol{1}_J'\boldsymbol{f}(\boldsymbol{X}_s))), \tag{4}$$

where $\boldsymbol{1}_J$ is a $J \times 1$ vector of ones. UN designs that maximize the logarithm of the determinant of the simplified information matrix in Equation (4), $\mathcal{D}_0(\boldsymbol{X}) = \log|\boldsymbol{M}(\boldsymbol{X}, \boldsymbol{0}_K)|$, are referred to as $\mathcal{D}_0$-optimal.

Locally optimal designs might provide imprecise estimates if the prior guess $\boldsymbol{\beta}_P$ is not close to the true value of the parameter vector $\boldsymbol{\beta}$. This weakness is overcome by the Bayesian design approach, which accounts for the prior uncertainty concerning the parameter values, and generally results in more precise estimates. The Bayesian design approach averages the value of the $\mathcal{D}$-optimality criterion over a prior distribution of likely parameter values, $\pi(\boldsymbol{\beta})$. It is customary to consider as a prior distribution $\pi(\boldsymbol{\beta})$, the multivariate normal distribution $\mathcal{N}(\boldsymbol{\beta}|\boldsymbol{\beta}_0, \boldsymbol{\Sigma}_0)$ with prior mean $\boldsymbol{\beta}_0$ and prior variance-covariance matrix $\boldsymbol{\Sigma}_0$. A Bayesian $\mathcal{D}$-optimal design maximizes the logarithm of the determinant of the information matrix averaged over the K-dimensional space $\mathcal{R}_k$ defined by the prior distribution

$$\mathcal{D}_B(\boldsymbol{X}) = \int_{\mathcal{R}_k} \log|\boldsymbol{M}(\boldsymbol{X}, \boldsymbol{\beta})|\pi(\boldsymbol{\beta})\, d\boldsymbol{\beta}. \tag{5}$$

In order to compare two different designs, we use the relative $\mathcal{D}$-efficiency as a quality measure. The $\mathcal{D}$-efficiency of a design $\boldsymbol{X}$ relative to a design $\boldsymbol{X}^*$ is defined as

$$\mathcal{D}\text{-eff}(\boldsymbol{X}, \boldsymbol{X}^*) = \exp\left(\frac{\mathcal{D}(\boldsymbol{X}) - \mathcal{D}(\boldsymbol{X}^*)}{K}\right), \tag{6}$$

where $\mathcal{D}$ corresponds to the $\mathcal{D}_0$-optimality criterion when comparing UN designs, to the $\mathcal{D}_P$-optimality criterion when comparing locally optimal designs, and to the $\mathcal{D}_B$-optimality criterion when comparing Bayesian designs.

3 Integrated algorithm

In this section, we describe the integrated algorithm for the optimal design of SC experiments with partial profiles. The algorithm requires an initial design as an input parameter. This initial design is constructed by randomly choosing the attributes to be held constant for each choice situation, and the levels of the varying attributes for each profile of the choice situation. The initial design is improved by modifying one choice situation at a time. In general terms, the algorithm improves the quality of the design by means of two procedures. The first procedure aims at finding the best set of attributes to be held constant in the choice situation. The second procedure aims at finding the best levels of the varying attributes. These two procedures are executed sequentially in order to find the configuration of the choice situation that produces the best value of the optimality criterion. The iterative process stops when no further improvement can be obtained for any of the design's choice situations.

Algorithm 1 shows the general layout of the integrated algorithm. The initial design $\boldsymbol{X}$ and the initial family of sets of constant attributes C are input parameters. All information concerning the number of choice situations, the number of profiles per choice situation and the profile strength is contained within $\boldsymbol{X}$ and C. The design $\boldsymbol{X}$ is composed of the design matrices $\boldsymbol{X}_s$ corresponding to each choice situation s, with $1 \leq s \leq S$. Similarly, C is composed of the sets of attributes C_s that are held constant in each choice situation s, with $1 \leq s \leq S$. For example, the set $C_1 = \{1, 2\}$ indicates that attributes 1 and 2 are held constant in choice situation 1. The function $\mathcal{D}$ calculates the value of the optimality criterion of a design. This value corresponds to the $\mathcal{D}_0$-optimality criterion when generating UN designs, to the $\mathcal{D}_P$-optimality criterion when generating locally optimal designs, and to the $\mathcal{D}_B$-optimality criterion when generating Bayesian designs. The variable $\mathcal{D}^0$ stores the value of the optimality criterion before the optimization process begins.

Algorithm 1: Integrated algorithm

```
Input: The initial design $\boldsymbol{X}$ and the initial family of sets of constant attributes $C$
$\mathcal{D}^0 \leftarrow \mathcal{D}(\boldsymbol{X})$
for $s \leftarrow 1$ to $S$ do
    improve_constant_attributes($\boldsymbol{X}, C, s$) // See Algorithm 3
    improve_varying_attributes($\boldsymbol{X}, C, s$) // See Algorithm 2
if $\mathcal{D}(\boldsymbol{X}) > \mathcal{D}^0$ then
    integrated_algorithm($\boldsymbol{X}$)
```

The varying attributes within a choice situation are improved using a modified version of the coordinate-exchange algorithm proposed by Meyer and Nachtsheim (1995). This algorithm, which was introduced in the SC design literature by Kessels et al. (2009), iteratively exchanges the level of each attribute for the one that produces the best value of the optimality criterion. It applies a first improvement strategy, which means that exchanges that lead to an improvement are performed as soon as they are found. The design $\boldsymbol{X}$, the family of sets of constant attributes C and the number of the choice situation s are input parameters of the coordinate-exchange algorithm. The pseudocode of this procedure is shown in Algorithm 2. In this algorithm, the number of levels of attribute f is denoted by L_f and the levels themselves are written as $1, 2, \ldots, L_f$. The variable $\boldsymbol{X}_{s,j,f}$ represents the level of attribute f in profile j of choice situation s. The variable $\mathcal{D}^*$ stores the best value of the optimality criterion found during the execution of the algorithm. An illustrative explanation of the coordinate-exchange algorithm for quantitative attributes can be found in Goos and Jones (2011).

Algorithm 2: Improving the levels of the varying factors (improve_varying_attributes)

```
Input: The design $\boldsymbol{X}$, the family of sets of constant attributes $C$ and the number of the choice
       situation, $s$, under consideration
$\mathcal{D}^0 \leftarrow \mathcal{D}(\boldsymbol{X})$
for $j \leftarrow 1$ to $J$ do
    for $f \leftarrow 1$ to $F$ do
        if $f \notin C_s$ then
            $\mathcal{D}^* \leftarrow \mathcal{D}(\boldsymbol{X})$
            $l^* \leftarrow \boldsymbol{X}_{s,j,f}$
            for $l \leftarrow 1$ to $L_f$ do
                $\boldsymbol{X}_{s,j,f} \leftarrow l$
                if $\mathcal{D}(\boldsymbol{X}) > \mathcal{D}^*$ then
                    $\mathcal{D}^* \leftarrow \mathcal{D}(\boldsymbol{X})$
                    $l^* \leftarrow l$
            $\boldsymbol{X}_{s,j,f} \leftarrow l^*$
if $\mathcal{D}(\boldsymbol{X}) > \mathcal{D}^0$ then
    improve_varying_attributes($\boldsymbol{X}, C$)
```

The set C_s of attributes that are held constant in choice situation s is improved one attribute at a time. At each iteration, one of the attributes is removed from the set (i.e., it is no longer considered constant) and treated as a varying attribute. Next, the algorithm evaluates the quality of the designs obtained by fixing one of the remaining attributes. All varying attributes are considered, and the one that produces the best value of the optimality criterion is added to the set of constant attributes. The algorithm finishes its execution when all the elements in the initial set of constant attributes have been evaluated for a potential exchange.

The pseudocode of the procedure to improve the set of constant attributes is shown in Algorithm 3. The goal of this algorithm is to find the best possible update of the family C of the sets of constant attributes $C_1, \ldots, C_S$. The updated version of the family C is denoted by $C^* = \{C_1^*, \ldots, C_S^*\}$. When an element c is removed from a set of constant attributes C_s, an additional improvement procedure is executed (line 6 in the pseudocode) to make the levels of the attribute different. This is done by applying the coordinate-exchange algorithm only to column c of the design matrix $\boldsymbol{X}_s$. Every time a varying attribute is considered for addition to C_s, it is necessary to store the levels for each profile in a temporary vector $\boldsymbol{T}$ (line 9 in the pseudocode). This is because, once an attribute is considered fixed and has been added to C_s, it is set to a constant level (line 11 in the pseudocode). The attribute level assigned has no effects on the estimation of the MNL model or the quality of the design. However, for consistency purposes, the algorithm always sets the attribute to its first level. The original levels of the attribute are restored (line 16 in the pseudocode) after the quality of the resulting design has been evaluated.

As shown in Algorithm 1, Algorithms 3 and 2 are executed sequentially in order to improve each choice situation of the design. Improving the set of constant attributes first, followed by improving the varying attributes, leads to the best algorithm performance. During the development phase of the algorithm, we

Algorithm 3: Improving the set of constant attributes (improve_constant_attributes)

```
Input: The design $\boldsymbol{X}$, the family of sets of constant attributes $C$ and the number of the choice
       situation, $s$, considered
$\mathcal{D}^* \leftarrow \mathcal{D}(\boldsymbol{X})$
$C^* \leftarrow C$
for $c \in C_s$ do
    $c^* \leftarrow c$
    $C^*_s \leftarrow C^*_s \setminus \{c\}$
    improve_varying_attributes($\boldsymbol{X}, C^*, s, c$) // Improve only the $c$-th column
    for $f \leftarrow 1$ to $F$ do
        if $f \notin C^*_s$ then
            $\boldsymbol{T}_J \leftarrow \boldsymbol{X}_{s,1:J,f}$
            $C^*_s \leftarrow C^*_s \cup \{f\}$
            $\boldsymbol{X}_{s,1:J,f} \leftarrow \mathbf{1}_J$
            if $\mathcal{D}(\boldsymbol{X}) > \mathcal{D}^*$ then
                $\mathcal{D}^* \leftarrow \mathcal{D}(\boldsymbol{X})$
                $c^* \leftarrow f$
            $C^*_s \leftarrow C^*_s \setminus \{f\}$
            $\boldsymbol{X}_{s,1:J,f} \leftarrow \boldsymbol{T}_J$
    $C^*_s \leftarrow C^*_s \cup \{c^*\}$
    $\boldsymbol{X}_{s,1:J,c^*} \leftarrow \mathbf{1}_J$
$C \leftarrow C^*$
```

observed that the choice of constant attributes has the largest impact on the quality of the design. Therefore, the algorithm generates better designs by making this decision first. We also tried other alternative improvement strategies. None of them, despite being more complex and computationally more demanding, outperformed the simple algorithm described. For example, we tested a more exhaustive improvement procedure for the set of constant attributes. Instead of improving one constant attribute at a time, we tried improving various constant attributes at a time. We also tried improving the levels of all varying attributes every time a new set of constant attributes is evaluated. In that way, the final set of constant attributes would be chosen considering the best configuration of the varying attributes. However, none of these improvement strategies generated designs with a better quality.

We recommend the integrated algorithm to be executed 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 (the one that optimizes the value of the optimality criterion). The multi-start strategy is common to the algorithms proposed by Huber and Zwerina (1996), Sándor and Wedel (2001), Kuhfeld and Tobias (2005), Kessels et al. (2009) and Quan et al. (2011). This strategy allows the algorithm to generate designs with a very good quality in a very short execution time.

When testing the algorithm on the utility-neutral benchmark instances described by Großmann et al. (2009), we noticed that, while it outperforms the algorithms of Kessels et al. (2011b) and Kessels et al. (2015), the algorithm is not able to find the optimal designs for some of the instances; not even when using a large number of restarts. The integrated algorithm generates designs that are equally efficient to those suggested by Großmann et al. (2009) for 28 out of 50 instances. For the other 22 instances, the integrated algorithm produces designs that are slightly less efficient (at most 0.44% less efficient) than the optimal designs. The results of this computational experiment are described in depth in Section 6.1 and shown in Table 1. The limitation of the integrated algorithm is due to the fact that each component of the algorithm improves the design by making locally optimal decisions. Since the algorithm improves one choice situation at a time, it is not able to evaluate the repercussions that these simple improvements have on the structure (and, hence, the quality) of the entire design. This is especially so when selecting the set of constant attributes for each choice situation, which is the decision with the largest impact.

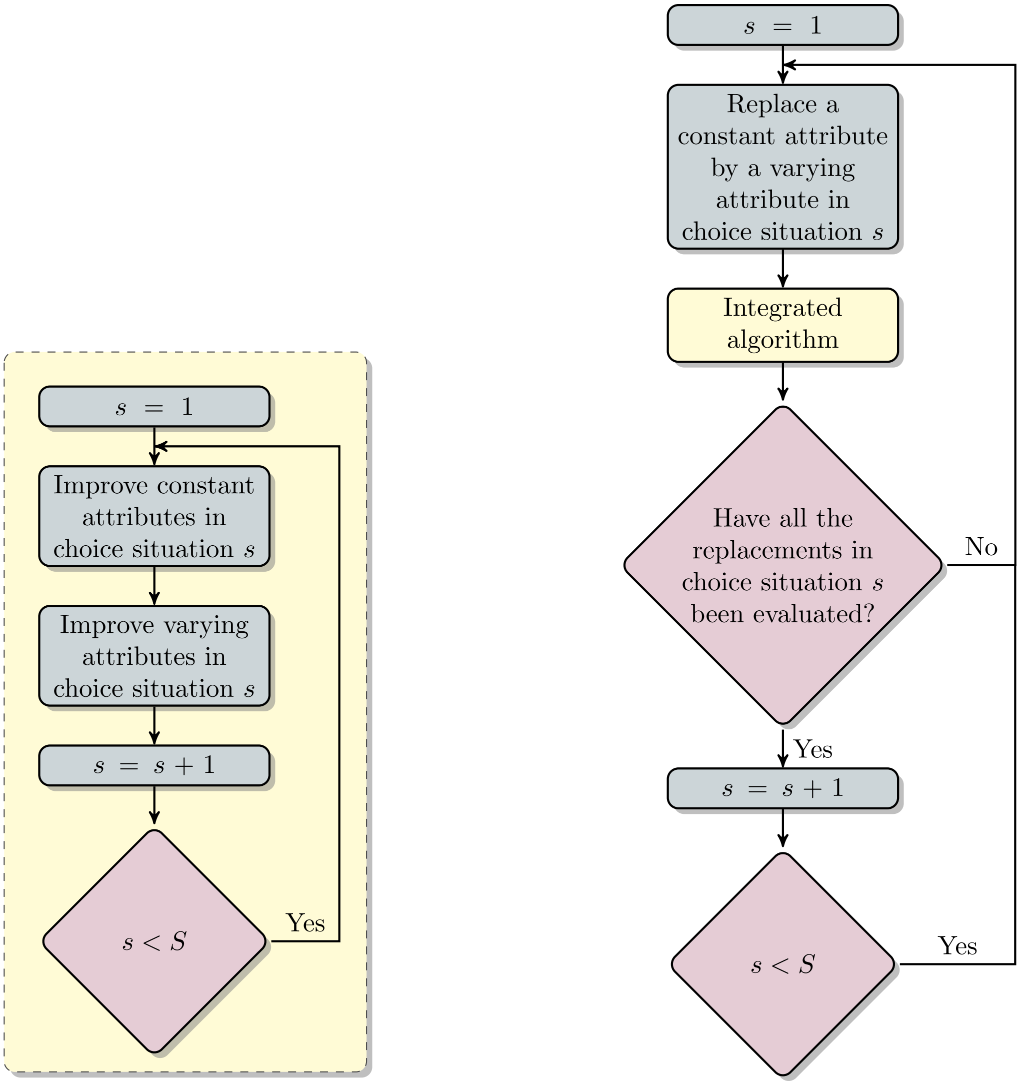
4 Exhaustive integrated algorithm

In order to overcome the limitation shown by the integrated algorithm, we propose an extra procedure that is executed on top of it. This additional procedure evaluates the effects that the selection of a set of constant attributes (for a particular choice situation) has on the entire design. This procedure iterates over the choice situations of the design and executes an algorithm similar to the one that improves the set of constant attributes of a choice situation (explained in Section 3 and shown in Algorithm 3). Every time a new configuration of constant attributes is evaluated in a certain choice situation, the algorithm executes the integrated algorithm for all other choice situations too in order to propagate the effects of this new configuration. This leads to an exponential increase in the complexity of this procedure compared to the integrated algorithm. However, this exhaustive approach performs a more thorough exploration of the design space and generates better designs. Due to this characteristic, we called this algorithm the *exhaustive integrated algorithm*. Figure 1 visualizes the workflows for one iteration of the integrated algorithm and one iteration of the exhaustive algorithm. In particular, it illustrates how the execution of the former is embedded in the latter.

The pseudocode of this new exhaustive algorithm is shown in Algorithm 4. Like Algorithm 3, Algorithm 4 aims at finding the best set of constant attributes. At each iteration, the algorithm evaluates the replacement of one element in the set of constant attributes by a varying attribute. Nonetheless, there are two important differences between Algorithms 3 and 4. First, in order to evaluate the quality of a replacement, Algorithm 4 creates extra copies $\boldsymbol{X}'$ and C' of $\boldsymbol{X}$ and C, respectively (lines 8 and 9 of the pseudocode). This is because, once a replacement has been carried out, the integrated algorithm is executed to improve the slightly modified design (line 14 of the pseudocode). By doing so, the effects of the change in the set of constant attributes of a given choice situation are propagated to the other choice situations of the design. The resulting design is the best one generated by the integrated algorithm if the new set of constant attributes evaluated is considered part of the initial structure. If this new design has a better quality than the best design found so far, it is stored in variables $\boldsymbol{X}^*$ and C^* (lines 15, 16 and 17 of the pseudocode). Note that it is possible that the resulting design has a very different structure compared to the slightly modified initial design. The second difference between Algorithms 3 and 4 is related to the number of exchanges evaluated by the algorithms when improving a choice situation. Observe that, in Algorithm 4, the original design is updated if a better design is found after evaluating the exchange of one element in the set of constant attributes (lines 18, 19 and 20 of the pseudocode). In such case, because the entire family of constant attributes potentially experienced a major change due to the nested execution of the integrated algorithm, it is necessary to restart the evaluation of the set C_s (the loop starting in line 3 of the pseudocode).

5 Implementation details

Both algorithms, the integrated algorithm from Section 3 and the exhaustive algorithm from Section 4, were implemented using C++. In order to speed up the execution of the algorithms, we paid special attention to the calculation of the optimality criteria. The matrix $\boldsymbol{X}_s$ of each choice situation s is separately stored in memory. By doing so, it is possible to implement update procedures that avoid recalculating the entire information matrix every time a choice situation is modified. When a new design $\boldsymbol{X}^*$ is generated by modifying choice situation s of a design $\boldsymbol{X}$, it is possible to calculate the corresponding information matrix



$s = 1$
Improve constant attributes in choice situation s
Improve varying attributes in choice situation s
$s = s + 1$
$s < S$
Yes
$s = 1$
Replace a constant attribute by a varying attribute in choice situation s
Integrated algorithm
Have all the replacements in choice situation s been evaluated?
No
Yes
$s = s + 1$
$s < S$
Yes

(a) Integrated algorithm

(b) Exhaustive integrated algorithm

Figure 1: Comparison between the execution of the integrated algorithm and the execution of the exhaustive integrated algorithm.

by using the formula:

$$\boldsymbol{M}(\boldsymbol{X}^*, \boldsymbol{\beta}) = \boldsymbol{M}(\boldsymbol{X}, \boldsymbol{\beta}) - \boldsymbol{f}'(\boldsymbol{X}_s)(\boldsymbol{P}_s - \boldsymbol{p}_s\boldsymbol{p}_s')\boldsymbol{f}(\boldsymbol{X}_s) + \boldsymbol{f}'(\boldsymbol{X}_s^*)(\boldsymbol{P}_s^* - \boldsymbol{p}_s^*\boldsymbol{p}_s^{*\prime})\boldsymbol{f}(\boldsymbol{X}_s^*), \tag{7}$$

where $\boldsymbol{P}_s^*$ and $\boldsymbol{p}_s^*$ are calculated using the matrix $\boldsymbol{X}_s^*$ of choice situation s in the new design $\boldsymbol{X}^*$. The new information matrix is calculated from the old one by subtracting the part of it corresponding to the old choice situation and adding the term corresponding to the new one (see Equations (3) and (4)). This strategy increases the memory used by the algorithms, but considerably reduces their execution times.

In order to calculate the K-dimensional integral in Equation (5) (related to the prior distribution of the parameter vector $\boldsymbol{\beta}$) when generating $\mathcal{D}_B$-optimal designs, the algorithm uses the integration method proposed by Gotwalt et al. (2009) and Gotwalt (2010). This method is based on the radial-spherical integration rule proposed by Monahan and Genz (1997). It transforms the integral in Equation (5) into a

Algorithm 4: Exhaustive algorithm

```
Input: The initial design $\boldsymbol{X}$ and the initial family of sets of constant attributes $C$
$\mathcal{D}^0 \leftarrow \mathcal{D}(\boldsymbol{X})$
for $s \leftarrow 1$ to $S$ do
    for $c \in C_s$ do
        $\boldsymbol{X}^* \leftarrow \boldsymbol{X}$
        $C^* \leftarrow C$
        for $f \leftarrow 1$ to $F$ do
            if $f \notin C_s$ then
                $\boldsymbol{X}' \leftarrow \boldsymbol{X}$
                $C' \leftarrow C$
                $C'_s \leftarrow C'_s \setminus \{c\}$
                improve_varying_attributes($X', C', s, c$)  // Improve only the $c$-th column
                $C'_s \leftarrow C'_s \cup \{f\}$
                $\boldsymbol{X}'_{s,1:J,f} \leftarrow \boldsymbol{1}_J$
                integrated_algorithm($\boldsymbol{X}', C'$)
                if $\mathcal{D}(\boldsymbol{X}') > \mathcal{D}(\boldsymbol{X}^*)$ then
                    $\boldsymbol{X}^* \leftarrow \boldsymbol{X}'$
                    $C^* \leftarrow C'$
        if $\boldsymbol{X} \neq \boldsymbol{X}^* \vee C \neq C^*$ then
            $\boldsymbol{X} \leftarrow \boldsymbol{X}^*$
            $C \leftarrow C^*$
if $\mathcal{D}(\boldsymbol{X}) > \mathcal{D}^0$ then
    exhaustive_algorithm($\boldsymbol{X}, C$)
```

radial component and a spherical surface component. The radial integration is performed using the generalized Gauss-Laguerre quadrature while the spherical integration is performed using a randomly rotated Mysovskikh (1980) extended simplex quadrature. Bliemer et al. (2008) and Yu et al. (2010) showed that quadrature methods usually outperform other methods for evaluating the Bayesian optimality criterion. Contrary to Monte Carlo sampling, quadrature methods estimate the integral by evaluating the values for the $\mathcal{D}_B$-optimality criterion using a small systematic sample of parameter vectors $\boldsymbol{\beta}$ from the prior distribution. In order to maximize the speed of our algorithm, the calculations related to each vector $\boldsymbol{\beta}$ in the sample are stored separately. This allows us to implement the update formula in Equation (7) for each sampled vector $\boldsymbol{\beta}$.

6 Computational experiments

In this section, we describe the results of several computational experiments we carried out with two main objectives. The first objective was to assess the performance of the algorithms under different experimental conditions. The second objective was to analyse the effects that these conditions have on the structure of the designs generated. Since there are many factors that could influence the algorithms' behaviour and the final structure of the designs, we executed four different experiments, each with a specific purpose. The first experiment was carried out in order to (i) obtain a general overview of the algorithms' performance and (ii) perform a first comparison to the latest version of two-stage algorithm (Kessels et al., 2015) and to the analytical generation method of Großmann et al. (2009). The purpose of the second experiment was to study the influence of the number of choice situations, the number of profiles per choice situation, the profile strength and the attributes' heterogeneity on the performance of the algorithms. The third experiment was carried out to study the effect of the parameter vector $\boldsymbol{\beta}$ when generating locally optimal designs. Finally, the purpose of the fourth experiment was to study the influence of the variance-covariance matrix of the prior distribution when generating Bayesian optimal designs. The experiments were executed using a computer with a 2.93 GHz Intel Core i7 processor. The two-stage algorithm of Kessels et al. (2015) and the integrated algorithm were executed using 1000 restarts, and the exhaustive (integrated) algorithm was executed using 25 restarts.

6.1 General overview and preliminary comparison

In order to obtain a general overview of the algorithms' performance, we ran the two-stage algorithm, the integrated algorithm and the exhaustive algorithm to generate UN designs for the 50 instances listed by Großmann et al. (2009). The results are shown in Table 1. The first five columns show information about the instances: the name, the number of attributes (F), the number of levels for each attribute, the profile strength (G) and the number of choice situations (S). The other columns show the $\mathcal{D}_0$-efficiencies of the best designs generated by each algorithm relative to the optimal designs analytically constructed by Großmann et al. (2009). Additionally, the average execution time of each restart is shown for the integrated and the exhaustive algorithm.

Table 1: Performance comparison of the two-stage, the integrated and the exhaustive algorithm using a benchmark set of 50 instances.

Instance information					Two-stage	Integrated		Exhaustive	
Name	F	Levels	G	S	$\mathcal{D}_0$-eff.	$\mathcal{D}_0$-eff.	Time (s)	$\mathcal{D}_0$-eff.	Time (s)
PP01	4	$2,3,3,3$	3	42	99.59	99.97	0.02	100.00	4.03
PP02	4	$2,2,3,3$	2	18	90.31	100.00	0.01	100.00	0.75
PP03	4	$2,2,3,3$	3	12	99.07	100.00	0.01	100.00	0.24
PP04	4	$2,2,4,4$	2	16	88.45	100.00	0.01	100.00	1.02
PP05	4	$2,2,4,4$	3	24	98.99	99.91	0.02	100.00	1.54
PP06	4	$2,2,5,5$	2	50	97.24	99.92	0.06	100.00	15.61
PP07	4	$2,2,5,5$	3	40	99.90	99.90	0.05	100.00	7.25
PP08	4	$3,3,4,4$	2	60	97.81	99.95	0.06	100.00	20.97
PP09	4	$3,3,5,5$	2	90	97.04	99.97	0.13	100.00	65.71
PP10	4	$2,2,2,3$	2	30	99.13	100.00	0.00	100.00	1.47
PP11	4	$2,2,2,3$	3	36	100.00	100.00	0.01	100.00	1.68
PP12	4	$2,2,2,4$	2	12	93.86	100.00	0.00	100.00	0.4
PP13	4	$2,2,2,4$	3	72	100.00	100.00	0.02	100.00	8.01
PP14	4	$2,2,2,5$	2	60	91.12	100.00	0.03	100.00	10.82
PP15	4	$3,3,3,4$	2	54	99.72	100.00	0.04	100.00	12.69
PP16	5	$2,3,3,3,3$	3	36	99.58	100.00	0.04	100.00	14.46
PP17	5	$2,2,3,3,3$	2	24	97.70	100.00	0.01	100.00	4.17
PP18	5	$2,2,3,3,3$	3	96	99.40	99.99	0.09	100.00	69.93
PP19	5	$2,2,4,4,4$	2	44	98.78	99.89	0.07	100.00	25.72
PP20	5	$2,2,5,5,5$	2	70	98.14	99.91	0.22	100.00	102.38
PP21	5	$2,2,2,3,3$	2	42	97.38	100.00	0.02	100.00	9.57
PP22	5	$2,2,2,3,3$	3	28	97.65	99.93	0.02	100.00	5.53
PP23	5	$2,2,2,4,4$	2	18	93.66	100.00	0.02	100.00	2.75
PP24	5	$2,2,2,4,4$	3	24	98.18	100.00	0.04	100.00	5.97
PP25	5	$3,3,3,4,4$	2	72	99.55	100.00	0.14	100.00	67.33
PP26	5	$3,3,3,4,4$	3	96	99.63	99.97	0.25	100.00	173.55

Table 1: Performance comparison of the two-stage, the integrated and the exhaustive algorithm using a benchmark set of 50 instances (continuation).

Instance information					Two-stage	Integrated		Exhaustive	
Name	F	Levels	G	S	$\mathcal{D}_0$-eff.	$\mathcal{D}_0$-eff.	Time (s)	$\mathcal{D}_0$-eff.	Time (s)
PP27	5	$2,2,2,2,3$	2	36	98.70	100.00	0.02	100.00	5.04
PP28	5	$2,2,2,2,3$	3	24	98.84	100.00	0.01	100.00	3.35
PP29	5	$2,2,2,2,4$	2	28	96.74	100.00	0.02	100.00	4.02
PP30	5	$2,2,2,2,4$	3	24	99.60	100.00	0.02	100.00	3.62
PP31	5	$2,2,2,2,5$	2	40	94.87	100.00	0.03	100.00	11.9
PP32	5	$2,2,2,2,5$	3	40	97.75	100.00	0.04	100.00	12.18
PP33	6	$2,2,3,3,3,3$	2	30	98.19	100.00	0.05	100.00	12.91
PP34	6	$2,2,4,4,4,4$	2	28	93.75	100.00	0.10	100.00	22.13
PP35	6	$2,2,5,5,5,5$	2	90	98.57	99.90	0.69	100.00	485.1
PP36	6	$3,3,4,4,4,4$	2	96	99.67	99.93	0.44	100.00	350.39
PP37	6	$2,2,2,3,3,3$	2	54	99.27	100.00	0.07	100.00	36.09
PP38	6	$2,2,2,3,3,3$	3	36	99.15	99.92	0.07	100.00	24.17
PP39	6	$2,2,2,4,4,4$	2	48	98.17	99.90	0.11	100.00	49.42
PP40	6	$2,2,2,4,4,4$	3	32	98.13	99.56	0.11	99.90	35.89
PP41	6	$2,2,2,5,5,5$	3	100	97.81	99.94	0.72	100.00	625.57
PP42	6	$2,2,2,2,3,3$	2	24	97.03	100.00	0.02	100.00	4.79
PP43	6	$2,2,2,2,3,3$	3	32	99.03	99.87	0.04	100.00	16.32
PP44	6	$2,2,2,2,4,4$	2	20	93.19	100.00	0.03	100.00	5.83
PP45	6	$2,2,2,2,4,4$	3	80	98.09	99.98	0.18	100.00	133.49
PP46	6	$2,2,2,2,5,5$	2	60	96.83	99.94	0.14	100.00	83.89
PP47	6	$2,2,2,2,5,5$	3	40	96.37	99.83	0.15	100.00	59.86
PP48	6	$3,3,3,3,4,4$	2	84	99.60	99.96	0.25	100.00	187.71
PP49	6	$2,2,2,2,2,4$	2	80	97.81	100.00	0.08	100.00	64.22
PP50	6	$2,2,2,2,2,5$	2	90	96.31	100.00	0.12	100.00	97.19
Average					97.51	99.96	0.10	99.99	59.37

The designs generated by the integrated algorithm and the exhaustive algorithm consistently outperform the designs generated by the two-stage algorithm. Moreover, the designs produced by the integrated algorithm have $\mathcal{D}_0$-efficiencies that are very close to 100%. The design with the lowest quality produced by the integrated algorithm, corresponding to instance PP40, is only 0.44% less efficient than the optimal design constructed by Großmann et al. (2009). Despite the fact that its execution time is very short, the integrated algorithm find optimal designs for 28 of the 50 instances. Its longest average execution time is less than one second. On the contrary, the exhaustive algorithm generates optimal designs for all the instances, except for instance PP40. To our knowledge, this is the first algorithm that is able to match the quality of the designs analytically derived. This effectiveness comes, however, at a high price: the execution time of the exhaustive algorithm is several orders of magnitude larger than the execution time of the simple integrated algorithm.

The two-stage algorithm generates designs with large $\mathcal{D}_0$-efficiencies for many of the experiments in the benchmark set. However, the $\mathcal{D}_0$-efficiencies of the designs generated by the integrated algorithm are usually larger. The difference in $\mathcal{D}_0$-efficiencies between the two algorithms is larger when the number of choice situations is small and all but one or two attributes have the same number of levels. For example, for instance PP14, involving three 2-level attributes and one 5-level attribute, the design generated by the two-stage algorithm is 8.88% less efficient than the one generated by the integrated algorithm.

The appendix provides additional information about the $\mathcal{D}_0$-efficiencies of the various restarts of the integrated and the exhaustive algorithm. For most of the instances, even the minimum $\mathcal{D}_0$-efficiencies achieved by the integrated and the exhaustive algorithm are larger than the maximum $\mathcal{D}_0$-efficiencies achieved by the two-stage algorithm. This shows the benefits of simultaneously optimizing the set of constant attributes and the levels of the varying attributes, when generating an optimal design for a SC experiment with partial profiles.

For some instances, the integrated algorithm and the exhaustive algorithm generate designs with very different structures than those proposed by Großmann et al. (2009). One such instance is PP23. Table 2 compares the designs generated for this instance by the different approaches. The attributes that are held constant in each choice situation are marked with an asterisk. The design analytically generated by Großmann et al. (2009) involves 12 choice situations where the three two-level attributes are held constant, and 6 choice situations where one two-level attribute and the two four-level attributes are held constant. The design generated by the integrated algorithm involves 6 choice situations where three two-level attributes are held constant, and 12 choice situations where two two-level attributes and one four-level attribute are held constant. The design generated by the exhaustive algorithm involves 8 choice situations where three two-level attributes are held constant, 8 choice situations where two two-level attributes and one four-level attribute are held constant, and 2 choice situations where one two-level attribute and the two four-level attributes are held constant. The existence of such designs, different from the ones generated by Großmann et al. (2009) but equally efficient, was hitherto unknown. The design generated by the two-stage algorithm is 6.34% less efficient than the other designs, and appears to have too few choice situations where three two-level attributes are held constant.

For some other instances, the designs produced by the integrated algorithm and the exhaustive algorithm are similar in structure to those produced by Großmann et al. (2009). Table 3 illustrates this for instance PP24, where the optimal designs generated by these approaches only hold the 2-level attributes constant. The design generated by the two-stage algorithm has a different structure (it also involves choice situations in which one 4-level attribute is held constant) and is 1.82% less efficient than the optimal design.

Table 2: UN designs for the instance PP23 involving 18 choice situations, 2 profiles per choice situation, three 2-level attributes, two 4-level attributes and a profile strength of 2.

CS	Großmann					Integrated					Exhaustive					Two-stage				
	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
1	*	*	*	1	1	*	*	*	1	1	*	*	*	1	2	*	*	*	2	1
	*	*	*	2	2	*	*	*	3	4	*	*	*	3	3	*	*	*	3	3
2	*	*	*	1	1	*	*	*	1	4	*	*	*	1	3	*	*	*	2	3
	*	*	*	3	3	*	*	*	3	1	*	*	*	3	2	*	*	*	4	1
3	*	*	*	1	1	*	*	*	2	1	*	*	*	2	1	*	*	*	3	1
	*	*	*	4	4	*	*	*	3	3	*	*	*	4	3	*	*	*	4	2
4	*	*	*	2	2	*	*	*	2	1	*	*	*	2	2	*	*	*	3	2
	*	*	*	3	3	*	*	*	4	2	*	*	*	3	4	*	*	*	4	3
5	*	*	*	2	2	*	*	*	2	2	*	*	*	2	3	*	*	1	*	2
	*	*	*	4	4	*	*	*	1	4	*	*	*	4	1	*	*	2	*	4
6	*	*	*	3	3	*	*	*	2	3	*	*	*	2	4	*	*	1	*	4
	*	*	*	4	4	*	*	*	3	1	*	*	*	3	2	*	*	2	*	2

Table 2: UN designs for the instance PP23 involving 18 choice situations, 2 profiles per choice situation, three 2-level attributes, two 4-level attributes and a profile strength of 2 (continuation).

CS	Großmann					Integrated					Exhaustive					Two-stage				
	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
7	*	*	*	1	2	*	*	1	*	3	*	*	*	3	1	*	*	1	1	*
	*	*	*	2	1	*	*	2	*	2	*	*	*	4	2	*	*	2	4	*
8	*	*	*	1	3	*	*	1	*	2	*	*	*	3	4	*	*	1	4	*
	*	*	*	3	1	*	*	2	*	3	*	*	*	1	2	*	*	2	1	*
9	*	*	*	1	4	*	*	1	3	*	*	*	1	*	1	*	1	*	*	3
	*	*	*	4	1	*	*	2	4	*	*	*	2	*	4	*	2	*	*	4
10	*	*	*	2	3	*	*	1	4	*	*	*	1	*	3	*	1	*	*	4
	*	*	*	3	2	*	*	2	3	*	*	*	2	*	4	*	2	*	*	3
11	*	*	*	2	4	*	1	*	*	3	*	*	1	*	4	*	1	*	1	*
	*	*	*	4	2	*	2	*	*	4	*	*	2	*	3	*	2	*	3	*
12	*	*	*	3	4	*	1	*	*	4	*	*	1	*	4	*	1	*	3	*
	*	*	*	4	3	*	2	*	*	3	*	*	2	*	1	*	2	*	1	*
13	*	1	1	*	*	*	1	*	1	*	*	1	*	1	*	1	*	*	*	4
	*	2	2	*	*	*	2	*	2	*	*	2	*	2	*	2	*	*	*	1
14	*	1	2	*	*	*	1	*	2	*	*	1	*	2	*	1	*	*	1	*
	*	2	1	*	*	*	2	*	1	*	*	2	*	1	*	2	*	*	2	*
15	1	*	1	*	*	1	*	*	1	*	1	*	*	1	*	1	*	*	2	*
	2	*	2	*	*	2	*	*	4	*	2	*	*	4	*	2	*	*	1	*
16	1	*	2	*	*	1	*	*	4	*	1	*	*	4	*	1	*	2	*	*
	2	*	1	*	*	2	*	*	1	*	2	*	*	1	*	2	*	1	*	*
17	1	1	*	*	*	1	*	*	*	2	1	1	*	*	*	1	1	*	*	*
	2	2	*	*	*	2	*	*	*	4	2	2	*	*	*	2	2	*	*	*
18	1	2	*	*	*	1	*	*	*	4	1	2	*	*	*	1	2	*	*	*
	2	1	*	*	*	2	*	*	*	2	2	1	*	*	*	2	1	*	*	*
$\mathcal{D}_0$-eff.			100.00					100.00					100.00					93.66		

Table 3: UN designs for the instance PP24 involving 24 choice situations, 2 profiles per choice situation, three 2-level attributes, two 4-level attributes and a profile strength of 3.

CS	Großmann					Integrated					Exhaustive					Two-stage				
	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
1	*	*	1	2	2	*	*	1	1	2	*	*	1	1	3	*	*	1	2	1
	*	*	2	4	4	*	*	2	2	3	*	*	2	3	4	*	*	2	4	4
2	*	*	1	2	4	*	*	1	2	3	*	*	1	1	4	*	*	1	2	4
	*	*	2	4	2	*	*	2	4	2	*	*	2	3	3	*	*	2	1	1
3	*	*	1	3	3	*	*	1	2	4	*	*	1	2	4	*	*	1	3	1
	*	*	2	4	4	*	*	2	3	2	*	*	2	4	1	*	*	2	1	3

Table 3: UN designs for the instance PP24 involving 24 choice situations, 2 profiles per choice situation, three 2-level attributes, two 4-level attributes and a profile strength of 3 (continuation).

CS	Großmann					Integrated					Exhaustive					Two-stage				
	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
4	*	*	1	3	4	*	*	1	3	1	*	*	1	3	1	*	*	1	4	2
	*	*	2	4	3	*	*	2	2	2	*	*	2	4	4	*	*	2	3	4
5	*	*	1	4	2	*	*	1	3	2	*	*	1	3	4	*	*	1	4	3
	*	*	2	2	4	*	*	2	1	4	*	*	2	2	3	*	*	2	1	4
6	*	*	1	4	3	*	*	1	3	2	*	*	1	3	4	*	*	1	4	4
	*	*	2	3	4	*	*	2	4	1	*	*	2	1	1	*	*	2	2	2
7	*	*	1	4	4	*	*	2	3	4	*	*	1	4	1	*	1	*	1	2
	*	*	2	2	2	*	*	1	4	1	*	*	2	1	2	*	2	*	3	1
8	*	*	1	4	4	*	*	1	4	4	*	*	1	4	2	*	1	*	2	4
	*	*	2	3	3	*	*	2	3	1	*	*	2	3	4	*	2	*	4	2
9	*	1	*	1	1	*	1	*	1	1	*	1	*	1	3	*	1	*	3	2
	*	2	*	4	4	*	2	*	3	2	*	2	*	3	2	*	2	*	1	1
10	*	1	*	1	4	*	1	*	1	2	*	1	*	1	4	*	1	*	3	3
	*	2	*	4	1	*	2	*	4	3	*	2	*	4	1	*	2	*	2	4
11	*	1	*	2	2	*	1	*	1	3	*	1	*	2	2	*	1	*	4	1
	*	2	*	3	3	*	2	*	2	4	*	2	*	1	4	*	2	*	2	3
12	*	1	*	2	3	*	1	*	2	2	*	1	*	2	2	*	1	*	4	3
	*	2	*	3	2	*	2	*	1	4	*	2	*	4	3	*	2	*	3	2
13	*	1	*	3	3	*	1	*	2	3	*	1	*	3	2	*	1	1	1	*
	*	2	*	2	2	*	2	*	3	1	*	2	*	2	4	*	2	2	4	*
14	*	1	*	4	4	*	1	*	3	1	*	1	*	3	3	*	1	2	2	*
	*	2	*	1	1	*	2	*	2	2	*	2	*	2	2	*	2	1	1	*
15	*	1	*	3	2	*	1	*	3	4	*	1	*	4	1	1	*	*	1	2
	*	2	*	2	3	*	2	*	1	1	*	2	*	3	2	2	*	*	2	1
16	*	1	*	4	1	*	1	*	4	4	*	1	*	4	4	1	*	*	2	2
	*	2	*	1	4	*	2	*	1	3	*	2	*	1	3	2	*	*	1	3
17	1	*	*	1	1	1	*	*	1	2	1	*	*	1	1	1	*	*	2	3
	2	*	*	2	2	2	*	*	3	4	2	*	*	4	2	2	*	*	3	2
18	1	*	*	1	1	1	*	*	1	4	1	*	*	1	2	1	*	*	3	3
	2	*	*	3	3	2	*	*	2	1	2	*	*	2	1	2	*	*	1	1
19	1	*	*	1	2	1	*	*	1	4	1	*	*	2	1	1	*	*	3	4
	2	*	*	2	1	2	*	*	4	1	2	*	*	4	4	2	*	*	2	2
20	1	*	*	1	3	1	*	*	2	1	1	*	*	2	3	1	*	*	4	4
	2	*	*	3	1	2	*	*	1	3	2	*	*	1	1	2	*	*	3	3
21	1	*	*	2	1	1	*	*	2	1	1	*	*	2	4	1	*	2	*	1
	2	*	*	1	2	2	*	*	4	3	2	*	*	1	2	2	*	1	*	3
22	1	*	*	2	2	1	*	*	3	3	1	*	*	3	1	1	*	1	1	*
	2	*	*	1	1	2	*	*	4	2	2	*	*	2	3	2	*	2	4	*

Table 3: UN designs for the instance PP24 involving 24 choice situations, 2 profiles per choice situation, three 2-level attributes, two 4-level attributes and a profile strength of 3 (continuation).

CS	Großmann					Integrated					Exhaustive					Two-stage				
	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
23	1	*	*	3	1	1	*	*	4	3	1	*	*	4	3	1	1	*	*	1
	2	*	*	1	3	2	*	*	1	4	2	*	*	2	1	2	2	*	*	4
24	1	*	*	3	3	1	*	*	4	4	1	*	*	4	2	1	2	*	*	3
	2	*	*	1	1	2	*	*	2	3	2	*	*	3	3	2	1	*	*	4
$\mathcal{D}_0$-eff.	100.00					100.00					100.00					98.18				

6.2 Influence of the experimental setup

The benchmark set used in the previous experiment includes a considerable number of instances. However, there are several experimental setups that are not considered. For example, the set does not include SC experiments with more than two profiles per choice situation. Since the algorithms that we propose are very general, a more structured set of instances that allows the evaluation of the algorithm in a wider range of situations was needed. Therefore, we identified the parameters that define a SC experiment and ran a full factorial experiment with the most interesting levels for these parameters. The parameters considered are the number of choice situations (S), the number of profiles per choice situation (J), the profile strength (G) and the attributes' heterogeneity in terms of the number of levels. The parameters and the corresponding levels are shown in Table 4. Observe that the attributes' heterogeneity is defined as a specific configuration of the numbers of attribute levels. The number of attributes was fixed to 6. It was not included as a factor in the study to limit the size of the experiment. The total number of instances considered in the factorial experiment was $2^3 \times 3 = 24$.

The two-stage algorithm, the integrated algorithm and the exhaustive algorithm were used to generate UN designs for each of the 24 instances in the factorial experiment. Since there exist no analytical results concerning optimal designs for these instances, the designs obtained by the exhaustive algorithm were considered optimal. The $\mathcal{D}_0$-efficiencies of the designs generated by the two-stage and the integrated algorithm were therefore calculated relative to those produced by the exhaustive algorithm. The results are shown in Table 5.

Table 4: Parameters and levels tested to study the influence of the experimental setup on the performance of the two-stage, the integrated and the exhaustive algorithm.

Factor	Levels
S	15, 30
J	2, 3
G	3, 4
Attributes' level heterogeneity	$(3,3,3,3,3,3)$, $(2,2,2,4,4,4)$, $(2,2,3,3,4,4)$

The results confirm that the integrated algorithm and the exhaustive algorithm generate designs with a better quality than those produced by the two-stage algorithm. In order to further study the influence of each factor on the quality of the designs generated by the two-stage algorithm and the integrated algorithm, two analyses of variance (ANOVA) were conducted using the $\mathcal{D}_0$-efficiency as a response. The p-values of the F-tests for the significance of each factor in the ANOVA models are shown in Table 6. The table shows the p-values for the the main effects only, since none of the interaction effects were significant.

Table 5: Performance comparison of the two-stage, the integrated and the exhaustive algorithm for the 24 instances in the factorial experiment described in Section 6.2.

Factors				Two-stage	Integrated		Exhaustive	
S	J	G	Num. of levels	$\mathcal{D}_0$-eff.	$\mathcal{D}_0$-eff.	Time (s)	$\mathcal{D}_0$-eff.	Time (s)
15	2	4	$3, 3, 3, 3, 3, 3$	99.18	99.34	0.04	100.00	6.77
15	2	4	$2, 2, 2, 4, 4, 4$	95.43	99.84	0.05	100.00	8.15
15	2	4	$2, 2, 3, 3, 4, 4$	98.02	99.19	0.05	100.00	7.65
15	2	3	$3, 3, 3, 3, 3, 3$	95.72	99.72	0.04	100.00	8.91
15	2	3	$2, 2, 2, 4, 4, 4$	95.03	98.92	0.05	100.00	9.00
15	2	3	$2, 2, 3, 3, 4, 4$	95.25	98.11	0.04	100.00	9.49
15	3	4	$3, 3, 3, 3, 3, 3$	99.69	99.66	0.07	100.00	10.15
15	3	4	$2, 2, 2, 4, 4, 4$	98.03	99.78	0.08	100.00	14.91
15	3	4	$2, 2, 3, 3, 4, 4$	98.64	99.71	0.08	100.00	14.49
15	3	3	$3, 3, 3, 3, 3, 3$	98.14	100.00	0.07	100.00	12.61
15	3	3	$2, 2, 2, 4, 4, 4$	97.61	99.79	0.08	100.00	17.11
15	3	3	$2, 2, 3, 3, 4, 4$	98.45	99.71	0.08	100.00	15.58
30	2	4	$3, 3, 3, 3, 3, 3$	99.75	99.81	0.09	100.00	27.10
30	2	4	$2, 2, 2, 4, 4, 4$	93.02	99.84	0.11	100.00	30.64
30	2	4	$2, 2, 3, 3, 4, 4$	98.61	99.71	0.10	100.00	29.71
30	2	3	$3, 3, 3, 3, 3, 3$	99.57	99.74	0.09	100.00	30.76
30	2	3	$2, 2, 2, 4, 4, 4$	97.01	99.95	0.10	100.00	29.86
30	2	3	$2, 2, 3, 3, 4, 4$	97.91	99.76	0.10	100.00	31.76
30	3	4	$3, 3, 3, 3, 3, 3$	99.85	99.91	0.15	100.00	38.6
30	3	4	$2, 2, 2, 4, 4, 4$	98.56	99.93	0.18	100.00	66.59
30	3	4	$2, 2, 3, 3, 4, 4$	98.71	99.91	0.17	100.00	54.5
30	3	3	$3, 3, 3, 3, 3, 3$	98.62	99.85	0.16	100.00	56.05
30	3	3	$2, 2, 2, 4, 4, 4$	98.35	99.92	0.17	100.00	85.86
30	3	3	$2, 2, 3, 3, 4, 4$	98.46	99.88	0.17	100.00	68.56
Average				97.82	99.67	0.10	100.00	28.53

Table 6: p-values of the F-tests for the significance of each factor in the ANOVA models.

Factor	Two-stage	Integrated
S	0.1570	**0.0147**
J	**0.0081**	**0.0221**
G	0.2537	0.4468
Attributes' level heterogeneity	**0.0064**	0.2481

The performance of the integrated algorithm is significantly influenced by the number of choice situations and the number of profiles per choice situation, but not by the profile strength or the heterogeneity in the attribute levels. The performance of the two-stage algorithm is significantly affected by the number of profiles per choice situation and the attributes' heterogeneity. Figure 2 visualizes the impact of the attributes' heterogeneity and shows the average $\mathcal{D}_0$-efficiencies of the designs generated by the two-stage algorithm and the integrated algorithm, for each configuration of the numbers of attribute levels. Observe that the performance of the two-stage algorithm is poorer when there are attributes with different numbers of levels.

This confirms the results from the preliminary performance comparison in Section 6.1.

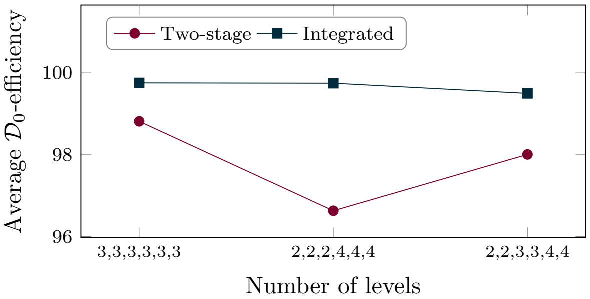


Figure 2: Influence of the attributes' heterogeneity (in terms of the number of levels) on the average $\mathcal{D}_0$-efficiencies of the designs generated by the two-stage and the integrated algorithm.

6.3 Influence of the prior parameter vector $\boldsymbol{\beta}$

The parameter vector $\boldsymbol{\beta}$ is another factor that might have an important influence on the design generation process. This vector directly affects the calculation of the optimality criteria through the probabilities in Equation (2). In the previous computational experiments, we set $\boldsymbol{\beta} = \mathbf{0}_K$ since various analytical results exist for that case. However, it is often more realistic to start from a different prior parameter vector. For example, it is known that people prefer short travel times over long ones, and low travel costs over high ones. This information can be included in the experimental design via the specification of the vector $\boldsymbol{\beta}$. By specifying part-worth values different from zero for a particular attribute, the experimenter can express his a priori belief about the respondents' preferences regarding the attribute's levels. This prior information might have an important impact on the structure of the design, especially on the number of times each of the attributes is held constant.

In order to study the effects of the prior parameter vector $\boldsymbol{\beta}$, we selected a fixed experimental configuration: a SC experiment with six 3-level attributes, 15 choice situations, 2 profiles per choice situation and a profile strength of 2. We also defined two factors in order to systematically generate the vectors $\boldsymbol{\beta}$ to be studied: the number of attributes the part-worths of which are different from zero (in other words, the number of attributes for which the experimenter has a priori knowledge of the respondents' preferences) and the magnitude of the part-worth values associated to those attributes (in other words, the experimenter's belief about the importance of those attributes and corresponding levels). In our study, we assumed ordinal attributes the levels of which are ordered from least preferred to most preferred, and we chose the non-zero part-worth values to be equally spaced around zero. This assumption, together with the use of effects-type coding, requires specifying the part-worth of the least preferred level only. Given a part-worth value i for the least preferred level, the part-worth values for the second and third levels are 0 and $-i$, respectively. In this experiment, we used three levels for the first factor and four levels for the second factor, resulting in $3 \times 4 = 12$ prior parameter vectors $\boldsymbol{\beta}$. The levels of each factor are shown in Table 7. Observe that the last level of the part-worth values (-2) is quite extreme, since it is not commonly observed in real experimental situations involving effects-type coded categorical attributes.

Using the two-stage algorithm, the integrated algorithm and the exhaustive algorithm, locally optimal designs were generated for each of the 12 nonzero parameter vectors $\boldsymbol{\beta}$ (defined by the factor levels in Table 7) and for $\boldsymbol{\beta} = \mathbf{0}_{12}$. Table 8 shows the $\mathcal{D}_P$-efficiencies of the designs obtained using each algorithm, together with the number of times each attribute is held constant.

Table 7: Factors and levels tested in Section 6.3 to study the influence of the prior parameter vector $\boldsymbol{\beta}$ on the performance of the two-stage, the integrated and the exhaustive algorithm.

Factor	Levels
Number of attributes with non-zero part-worth values	$2, 4, 6$
Part-worth values of the least preferred level	$-0.5, -1, -1.5, -2$

The two-stage algorithm generates designs in which all the attributes are held constant an equal number of times. This is because the prior parameter vector $\boldsymbol{\beta}$ is not taken into account during the first stage of the algorithm, where the constant attributes are defined for each choice situation. In the designs generated by the integrated and the exhaustive algorithm, the attributes with non-zero prior part-worth values are held constant more frequently. The larger the magnitude of the part-worth values, the larger the number of times these attributes are held constant. This phenomenon is more pronounced when there are fewer attributes with non-zero prior part-worth values. The lack of balance in the number of times attributes are held constant can be explained as follows: when the part-worth values of an attribute are large, the choice probabilities for profiles that differ with respect to that attribute are extreme. Therefore, choice situations for which that attribute is not held constant are not considered very informative. As a result, such choice situations are not selected by the $\mathcal{D}$-optimality criterion.

The lack of balance in the number of times the attributes are held constant in the designs produced by the integrated and the exhaustive algorithm demonstrates an important relationship with the performance of the two-stage algorithm. The larger the lack of balance in the pattern of constant attributes in the designs produced by the integrated algorithms, the poorer the performance of the designs produced by the two-stage algorithms. For example, when there are only two attributes with non-zero prior part-worth values and the part-worth of the least preferred level is equal to -1.5, the design generated by the two-stage algorithm is 8.91% less efficient than that generated by the integrated algorithm. This illustrates the benefits of considering the prior parameter vector $\boldsymbol{\beta}$ during the optimization of the set of constant attributes while generating locally optimal designs.

Table 8: Performance comparison of the two-stage, the integrated and the exhaustive algorithm for 13 prior parameter vectors $\boldsymbol{\beta}$ described in Section 6.3.

	Two-stage		Integrated		Exhaustive	
$\boldsymbol{\beta}$	$\mathcal{D}_P$-eff.	Num. const.	$\mathcal{D}_P$-eff.	Num. const.	$\mathcal{D}_P$-eff.	Num. const.
0 0 0 0 0 0 0 0 0 0 0 0	99.42	5 5 5 5 5 5	99.42	6 6 4 4 5 5	100.00	5 5 6 6 4 4
−0.5 0 −0.5 0 0 0 0 0 0 0 0 0	98.35	5 5 5 5 5 5	99.50	5 7 4 5 5 4	100.00	6 6 4 5 4 5
−0.5 0 −0.5 0 −0.5 0 −0.5 0 0 0 0 0	99.83	5 5 5 5 5 5	99.83	5 5 6 6 4 4	100.00	5 5 6 6 4 4
−0.5 0 −0.5 0 −0.5 0 −0.5 0 −0.5 0 −0.5 0	99.34	5 5 5 5 5 5	99.34	5 5 5 5 5 5	100.00	4 5 5 5 6 5
−1.0 0 −1.0 0 0 0 0 0 0 0 0 0	96.40	5 5 5 5 5 5	99.09	7 7 4 3 5 4	100.00	8 7 4 4 4 3
−1.0 0 −1.0 0 −1.0 0 −1.0 0 0 0 0 0	98.18	5 5 5 5 5 5	99.42	7 4 7 5 3 4	100.00	6 5 6 6 4 3
−1.0 0 −1.0 0 −1.0 0 −1.0 0 −1.0 0 −1.0 0	97.78	5 5 5 5 5 5	98.35	5 4 5 6 4 6	100.00	5 6 5 4 4 6
−1.5 0 −1.5 0 0 0 0 0 0 0 0 0	91.01	5 5 5 5 5 5	99.92	8 7 4 3 4 4	100.00	7 6 5 4 4 4
−1.5 0 −1.5 0 −1.5 0 −1.5 0 0 0 0 0	97.45	5 5 5 5 5 5	97.94	5 7 7 5 3 3	100.00	5 7 6 5 4 3
−1.5 0 −1.5 0 −1.5 0 −1.5 0 −1.5 0 −1.5 0	97.45	5 5 5 5 5 5	96.64	4 5 4 5 6 6	100.00	6 4 3 5 6 6
−2.0 0 −2.0 0 0 0 0 0 0 0 0 0	81.94	5 5 5 5 5 5	99.67	9 7 3 4 3 4	100.00	7 9 3 4 2 5
−2.0 0 −2.0 0 −2.0 0 −2.0 0 0 0 0 0	97.21	5 5 5 5 5 5	97.21	6 7 5 6 2 4	100.00	6 7 6 5 4 2
−2.0 0 −2.0 0 −2.0 0 −2.0 0 −2.0 0 −2.0 0	96.56	5 5 5 5 5 5	96.56	5 4 4 5 6 6	100.00	4 4 6 6 5 5
Average	96.22		98.68		100.00	

6.4 Influence of the model parameters' prior variance

Bayesian optimal designs are based on a prior distribution of parameter values with mean vector $\boldsymbol{\beta}_0$ and variance-covariance matrix $\boldsymbol{\Sigma}_0$. The larger the variance specified for a parameter, the larger the experimenter's degree of uncertainty about its value. In order to study the influence of the prior variance, we conducted an experiment with characteristics similar to the one described in Section 6.3. We used the same design configuration, and a fixed vector of prior means, $\boldsymbol{\beta}_0 = \mathbf{0}_{12}$. We also defined two factors in order to generate the variance-covariance matrices $\boldsymbol{\Sigma}_0$ to be studied: the number of attributes with non-zero variances for the parameters (in other words, the number of attributes for which the experimenter is uncertain about the true part-worth values) and the magnitude of the variance of the parameters. We considered three levels for the first factor and four levels for the second factor, and hence, $3 \times 4 = 12$ different configurations of the variance matrix $\boldsymbol{\Sigma}_0$. The levels for each factor are shown in Table 9. In order to obtain equal variances for all the parameters, we followed the recommendations provided by Kessels et al. (2008). For each attribute f, we specified negative covariances between its $L_f - 1$ part-worth values corresponding to a correlation coefficient of $-1/(L_f - 1)$.

Table 9: Factors and levels tested to study the influence of the model parameters' prior variance on the performance of the two-stage, the integrated and the exhaustive algorithm.

Parameter	Levels
Number of attributes with non-zero variance	$2, 4, 6$
Variance	$0.1, 0.2, 0.3, 0.4$

The two-stage algorithm, the integrated algorithm and the exhaustive algorithm were executed in order to generate Bayesian optimal designs using each of the 12 variance-covariance matrices $\boldsymbol{\Sigma}_0$ generated. An additional control instance was included in the study for which all parameters had a variance equal to zero. This instance assumes complete certainty about the parameter values, and the design generated is equivalent to the UN design. Table 10 shows the $\mathcal{D}_B$-efficiencies of the designs obtained by each algorithm.

The integrated and the exhaustive algorithm generate designs in which the attributes with a large prior variance are held constant more often. The larger the prior variance of the parameters, the larger the number of times the attributes are held constant. This difference is more prominent when the number of attributes with non-zero prior parameter variances is small. These results are in line with those discussed by Kessels et al. (2011a,c). The unbalance is a result of the sampling procedure from the normal distribution of parameter values. A prior distribution with large variances for some parameters covers some scenarios where there are strong preferences for certain attribute levels. The larger the variance of the distribution, the larger some of the parameter values sampled and the larger the part-worth values. As explained in Section 6.3, attributes with large part-worth values lead to many choice situations for which these attributes are held constant. For this reason, the design generation process holds attributes with large prior variances for its parameters constant more often.

As previously stated, the prior variance is a measure of uncertainty about the true values of the parameters associated with an attribute. For that reason, the insight previously described is somehow counterintuitive. The larger the uncertainty about an attribute, the more information the experimenter hopes to collect about that attribute and thus the lower the number of times the attribute may be expected to be held constant in the design. Nevertheless, the computational experiments carried out show the opposite. This is due to the MNL formula used to calculate the probability that a respondent chooses a profile, in combination with the sampling mechanism of the Bayesian approach.

Table 10: Performance comparison of the two-stage, the integrated and the exhaustive algorithm using 13 prior variance specifications for the parameters.

	Two-stage		Integrated		Exhaustive	
Variances	$\mathcal{D}_B$-eff.	Num. const.	$\mathcal{D}_B$-eff.	Num. const.	$\mathcal{D}_B$-eff.	Num. const.
0 0 0 0 0 0	99.17	5 5 5 5 5 5	99.25	5 5 5 6 4 5	100.00	5 5 6 6 4 4
0.1 0.1 0 0 0 0	98.43	5 5 5 5 5 5	99.75	6 7 6 4 3 4	100.00	6 6 4 5 5 4
0.1 0.1 0.1 0.1 0 0	98.84	5 5 5 5 5 5	99.75	6 5 6 6 4 3	100.00	4 5 7 7 4 3
0.1 0.1 0.1 0.1 0.1 0.1	98.84	5 5 5 5 5 5	99.75	6 4 6 4 5 5	100.00	5 5 5 5 5 5
0.2 0.2 0 0 0 0	97.29	5 5 5 5 5 5	99.67	7 7 3 5 4 4	100.00	8 7 4 3 4 4
0.2 0.2 0.2 0.2 0 0	97.04	5 5 5 5 5 5	99.50	6 6 6 6 2 4	100.00	6 7 6 6 2 3
0.2 0.2 0.2 0.2 0.2 0.2	98.84	5 5 5 5 5 5	99.34	6 4 5 5 5 5	100.00	6 4 5 5 5 5
0.3 0.3 0 0 0 0	94.96	5 5 5 5 5 5	99.34	8 9 4 2 3 4	100.00	9 9 3 2 3 4
0.3 0.3 0.3 0.3 0 0	95.92	5 5 5 5 5 5	99.42	7 7 6 7 1 2	100.00	7 7 7 6 0 3
0.3 0.3 0.3 0.3 0.3 0.3	99.42	5 5 5 5 5 5	99.34	5 5 5 5 5 5	100.00	5 5 4 6 5 5
0.4 0.4 0 0 0 0	93.16	5 5 5 5 5 5	99.83	9 9 4 2 3 3	100.00	9 9 2 4 3 3
0.4 0.4 0.4 0.4 0 0	93.94	5 5 5 5 5 5	99.42	7 7 7 7 1 1	100.00	8 7 7 7 0 1
0.4 0.4 0.4 0.4 0.4 0.4	99.09	5 5 5 5 5 5	99.00	5 5 5 4 6 5	100.00	5 5 6 3 5 6
Average	97.30		99.49		100.00	

7 Proof-of-concept example

In this section, we revisit the SC experiment carried out by Kupfer et al. (2013) in order to show the benefits of using the integrated algorithm in a real-life scenario. In this paper, the authors study the airport competition for air cargo in Europe. They identify the most important characteristics that airlines take into account when selecting an airport for their freighter operations, and thereby offer better insights to airports and policy makers about how to attract new service providers. Table 11 shows the airport characteristics considered as attributes in the SC experiment, along with their numbers of levels. The preferences of the airlines were collected by means of questionnaires consisting of 20 choice situations, each involving two alternative airports as profiles. In order to limit the cognitive burden imposed on the respondents, each choice situation showed only four out of the six attributes studied. In other words, each choice situation was composed of partial profiles in which two attributes were held constant.

Table 11: Attributes studied in the SC experiment of Kupfer et al. (2013).

Identifier	Attribute	Number of levels
1	Night-time restrictions	3
2	Airport experience with cargo	3
3	Presence of forwarders	3
4	Presence of passenger airlines	3
5	Airport charges (including handling)	5
6	Origin-destination demand	5

Kupfer et al. (2013) expected the origin-destination demand (attribute 6) to dominate the airlines' choices. For that reason, their SC experiment involved choice situations in which attribute 6 was held constant more often. To achieve that, the authors generated two different Bayesian $\mathcal{D}$-optimal designs using the two-stage algorithm of Kessels et al. (2015): a 6-attribute design involving 40 choice situations and two constant attributes, and a 5-attribute design (excluding attribute 6) involving 40 choice situations and one constant attribute. Both designs consist of four different blocks or surveys of 10 choice situations, where one block from the 6-attribute design and one block from the 5-attribute design were assigned to each respondent. In doing so, the authors ensured that attribute 6 was held constant in at least 50% of the choice situations that composed the resulting four questionnaires.

The design for the previous SC experiment can be improved by using the integrated algorithm. First, as shown in Sections 6.3 and 6.4, the integrated algorithm generates Bayesian $\mathcal{D}$-optimal designs than are more efficient that those generated by the two-stage algorithm. In fact, the 5-attribute and the 6-attribute designs generated by the two-stage algorithm (considering the same prior distribution of parameter values) are between 1% and 2% less $\mathcal{D}_B$-efficient than the designs generated by the integrated algorithm. This is mainly due to the ability of the integrated algorithm to generate designs with a better choice for the constant attributes. The differences between these designs are reported in Table 12, which shows the number of times each attribute is held constant in each design. Observe that the differences are more substantial between the 6-attribute designs.

Table 12: Number of times each attribute is held constant in the designs generated by the two-stage algorithm and the designs generated by the integrated algorithm for the SC experiment of Kupfer et al. (2013).

a) 6-attribute designs

	Attribute					
Algorithm	1	2	3	4	5	6
Two-stage	16	16	17	17	7	7
Integrated	21	20	17	19	0	3

b) 5-attribute designs

	Attribute				
Algorithm	1	2	3	4	5
Two-stage	10	10	10	10	0
Integrated	11	10	11	8	0

A second advantage of using the integrated algorithm is that an ad hoc design construction, involving two different subdesigns, is no longer needed. If an experimenter expects an attribute to be dominant (as was the case in Kupfer et al. (2013)), and would like to have a design in which that particular attribute is held constant more often, it is not necessary to combine two different designs in an ad hoc fashion. The experimenter can express his/her a priori belief by specifying larger part-worth values for that particular attribute (reflecting the attribute's dominance) and generating a Bayesian optimal design: the integrated algorithm will then automatically generate a design in which this attribute is held constant more often.

8 Summary and conclusion

In this paper, we have dealt with the generation of $\mathcal{D}$-optimal designs for SC experiments with partial profiles. The theoretical approaches available in the literature allow the generation of optimal designs for a restricted set of experimental situations. In response to these limitations, Kessels et al. (2011b, 2015) proposed more flexible two-stage algorithms. The first stage of the algorithms determines the set of constant attributes for each choice situation, and the second stage determines the levels of the varying attributes. These algorithms, however, show some limitations due to the fact that two optimization steps are carried out independently. In this paper, we propose an integrated algorithm that determines the set of constant attributes and the levels of the varying attributes simultaneously. The algorithm modifies the design, choice situation by choice situation, in order to optimize the value of the design optimality criterion. The integrated algorithm, together with a simple multi-start strategy, generates designs with a very good quality in a short execution time. Because the algorithm does not always find the optimal designs, we also propose an extension to the integrated algorithm. This extension leads to a more exhaustive algorithm with a considerably longer execution time. However, it does find the optimal designs for 98% of the benchmark experiments considered. As a result, the exhaustive algorithm is very powerful besides being very flexible.

The extensive set of computational experiments carried out shows the benefits of performing a simultaneous optimization. The designs generated by the integrated algorithm and the exhaustive version consistently outperform those generated by the two-stage algorithm. This is mainly because the first stage of the two-stage algorithm, which defines the structure of constant attributes, does not consider all the characteristics of the experiment. This leads the two-stage algorithm to generate designs with structures that are sometimes very different from those of the true optimal designs. This performance difference is even more prominent when creating locally optimal and Bayesian optimal designs. The $\mathcal{D}_B$-optimality criterion leads to designs for which the attributes with large prior values for the part-worths or large prior variances are held constant more often. The former result should not come as a surprise, since large prior values for the part-worths indicate that the experimenter is certain about strong preferences regarding these attributes. Therefore, it is reasonable to hold these attributes constant more often. However, the latter result is counterintuitive. By specifying a large prior variance for an attribute's part-worths, the experimenter expresses a large degree of uncertainty about the true values of the attribute's part-worths. Therefore, the experimenter certainly hopes that attribute to be held constant less often in order to collect more information about it. These results cast some doubt about the way in which the Bayesian $\mathcal{D}$-optimality criterion is implemented in the literature on SC experiments, and clearly encourage the study of alternative criteria in future research.

9 Acknowledgement

We acknowledge the financial support of the Flemish Fund for Scientific Research (FWO).

A Appendix: General overview and preliminary comparison (extended results)

Table 13: Detailed performance comparison of the two-stage, the integrated and the exhaustive algorithm using a benchmark set of 50 instances.

					Two-stage	Integrated				Exhaustive			
Instance information					$\mathcal{D}_0$-eff.	$\mathcal{D}_0$-efficiency			Time (s)	$\mathcal{D}_0$-efficiency			Time (s)
Name	F	Levels	G	S	Max.	Min.	Avg.	Max.	Avg.	Min.	Avg.	Max.	Avg.
PP01	4	$2,3,3,3$	3	42	99.59	99.70	99.86	99.97	0.02	99.96	99.97	100.00	4.03
PP02	4	$2,2,3,3$	2	18	90.31	97.84	99.29	100.00	0.01	100.00	100.00	100.00	0.75
PP03	4	$2,2,3,3$	3	12	99.07	95.78	98.30	100.00	0.01	99.53	99.81	100.00	0.24
PP04	4	$2,2,4,4$	2	16	88.45	96.31	98.39	100.00	0.01	99.20	99.70	100.00	1.02
PP05	4	$2,2,4,4$	3	24	98.99	98.82	99.48	99.91	0.02	99.72	99.85	100.00	1.54
PP06	4	$2,2,5,5$	2	50	97.24	99.39	99.73	99.92	0.06	99.90	99.96	100.00	15.61
PP07	4	$2,2,5,5$	3	40	99.90	99.39	99.68	99.90	0.05	99.85	99.92	100.00	7.25
PP08	4	$3,3,4,4$	2	60	97.81	99.61	99.82	99.95	0.06	99.95	99.99	100.00	20.97
PP09	4	$3,3,5,5$	2	90	97.04	99.78	99.88	99.97	0.13	99.97	99.99	100.00	65.71
PP10	4	$2,2,2,3$	2	30	99.13	99.46	99.86	100.00	0.00	99.72	99.99	100.00	1.47
PP11	4	$2,2,2,3$	3	36	100.00	99.11	99.90	100.00	0.01	99.90	99.99	100.00	1.68
PP12	4	$2,2,2,4$	2	12	93.86	95.65	98.17	100.00	0.00	100.00	100.00	100.00	0.4

Table 13: Detailed performance comparison of the two-stage, the integrated and the exhaustive algorithm using a benchmark set of 50 instances (continuation).

					Two-stage	Integrated				Exhaustive			
Instance information					$\mathcal{D}_0$-eff.	$\mathcal{D}_0$-efficiency			Time (s)	$\mathcal{D}_0$-efficiency			Time (s)
Name	F	Levels	G	S	Max.	Min.	Avg.	Max.	Avg.	Min.	Avg.	Max.	Avg.
PP13	4	2, 2, 2, 4	3	72	100.00	99.81	99.96	100.00	0.02	99.98	99.90	100.00	8.01
PP14	4	2, 2, 2, 5	2	60	91.12	99.79	99.93	100.00	0.03	100.00	100.00	100.00	10.82
PP15	4	3, 3, 3, 4	2	54	99.72	99.59	99.81	100.00	0.04	99.92	99.98	100.00	12.69
PP16	5	2, 3, 3, 3, 3	3	36	99.58	99.35	99.66	100.00	0.04	99.86	99.91	100.00	14.46
PP17	5	2, 2, 3, 3, 3	2	24	97.70	98.23	99.34	100.00	0.01	99.74	99.95	100.00	4.17
PP18	5	2, 2, 3, 3, 3	3	96	99.40	99.93	99.97	99.99	0.09	99.99	99.99	100.00	69.93
PP19	5	2, 2, 4, 4, 4	2	44	98.78	99.19	99.58	99.89	0.07	99.86	99.93	100.00	25.72
PP20	5	2, 2, 5, 5, 5	2	70	98.14	99.52	99.73	99.91	0.22	99.93	99.96	100.00	102.38
PP21	5	2, 2, 2, 3, 3	2	42	97.38	99.47	99.82	100.00	0.02	99.93	99.99	100.00	9.57
PP22	5	2, 2, 2, 3, 3	3	28	97.65	99.07	99.63	99.93	0.02	99.83	99.92	100.00	5.53
PP23	5	2, 2, 2, 4, 4	2	18	93.66	96.02	98.59	100.00	0.02	99.29	99.84	100.00	2.75
PP24	5	2, 2, 2, 4, 4	3	24	98.18	98.08	99.06	100.00	0.04	99.48	99.83	100.00	5.97
PP25	5	3, 3, 3, 4, 4	2	72	99.55	99.64	99.82	100.00	0.14	99.96	99.98	100.00	67.33
PP26	5	3, 3, 3, 4, 4	3	96	99.63	99.86	99.92	99.97	0.25	99.97	99.99	100.00	173.55
PP27	5	2, 2, 2, 2, 3	2	36	98.70	99.23	99.79	100.00	0.02	99.77	99.98	100.00	5.04
PP28	5	2, 2, 2, 2, 3	3	24	98.84	97.08	99.36	100.00	0.01	99.28	99.66	100.00	3.35
PP29	5	2, 2, 2, 2, 4	2	28	96.74	98.71	99.59	100.00	0.02	99.78	99.99	100.00	4.02
PP30	5	2, 2, 2, 2, 4	3	24	99.60	97.90	99.31	100.00	0.02	99.43	99.75	100.00	3.62
PP31	5	2, 2, 2, 2, 5	2	40	94.87	99.11	99.70	100.00	0.03	100.00	100.00	100.00	11.9
PP32	5	2, 2, 2, 2, 5	3	40	97.75	98.55	99.62	100.00	0.04	99.65	99.86	100.00	12.18
PP33	6	2, 2, 3, 3, 3, 3	2	30	98.19	98.10	99.32	100.00	0.05	99.72	99.91	100.00	12.91
PP34	6	2, 2, 4, 4, 4, 4	2	28	93.75	96.60	98.44	100.00	0.10	99.32	99.89	100.00	22.13
PP35	6	2, 2, 5, 5, 5, 5	2	90	98.57	99.53	99.71	99.90	0.69	99.92	99.97	100.00	485.1
PP36	6	3, 3, 4, 4, 4, 4	2	96	99.67	99.70	99.81	99.93	0.44	99.97	99.99	100.00	350.39
PP37	6	2, 2, 2, 3, 3, 3	2	54	99.27	99.57	99.82	100.00	0.07	99.94	99.98	100.00	36.09
PP38	6	2, 2, 2, 3, 3, 3	3	36	99.15	99.30	99.64	99.92	0.07	99.86	99.92	100.00	24.17
PP39	6	2, 2, 2, 4, 4, 4	2	48	98.17	99.18	99.59	99.90	0.11	99.87	99.95	100.00	49.42
PP40	6	2, 2, 2, 4, 4, 4	3	32	98.13	98.49	99.08	99.56	0.11	99.51	99.67	99.90	35.89
PP41	6	2, 2, 2, 5, 5, 5	3	100	97.81	99.79	99.88	99.94	0.72	99.95	99.97	100.00	625.57
PP42	6	2, 2, 2, 2, 3, 3	2	24	97.03	97.61	99.34	100.00	0.02	100.00	100.00	100.00	4.79
PP43	6	2, 2, 2, 2, 3, 3	3	32	99.03	99.17	99.60	99.87	0.04	99.83	99.91	100.00	16.32
PP44	6	2, 2, 2, 2, 4, 4	2	20	93.19	96.59	98.69	100.00	0.03	99.52	99.90	100.00	5.83
PP45	6	2, 2, 2, 2, 4, 4	3	80	98.09	99.83	99.92	99.98	0.18	99.97	99.99	100.00	133.49
PP46	6	2, 2, 2, 2, 5, 5	2	60	96.83	99.46	99.73	99.94	0.14	99.92	99.95	100.00	83.89
PP47	6	2, 2, 2, 2, 5, 5	3	40	96.37	98.78	99.40	99.83	0.15	99.61	99.87	100.00	59.86
PP48	6	3, 3, 3, 3, 4, 4	2	84	99.60	99.67	99.81	99.96	0.25	99.96	99.97	100.00	187.71
PP49	6	2, 2, 2, 2, 2, 4	2	80	97.81	99.73	99.93	100.00	0.08	100.00	100.00	100.00	64.22
PP50	6	2, 2, 2, 2, 2, 5	2	90	96.31	99.81	99.93	100.00	0.12	100.00	100.00	100.00	97.19
Average					97.51	98.80	99.52	99.96	0.10	99.80	99.93	99.99	59.37

References

Aoife A. Ahern and Nigel Tapley. The use of stated preference techniques to model modal choices on interurban trips in Ireland. *Transportation Research Part A: Policy and Practice*, 42(1):15–27, 2008.

Christopher M. Anderson, Chandita Das, and Timothy J. Tyrrell. Parking preferences among tourists in Newport, Rhode Island. *Transportation Research Part A: Policy and Practice*, 40(4):334–353, 2006.

Rob J. I. Basten, Matthieu C. Van der Heijden, and J. Marco J. Schutten. Joint optimization of level of repair analysis and spare parts stocks. *European Journal of Operational Research*, 222(3):474–483, 2012.

Chandra R. Bhat and Rupali Sardesai. The impact of stop-making and travel time reliability on commute mode choice. *Transportation Research Part B: Methodological*, 40(9):709–730, 2006.

Michiel C. J. Bliemer and John M. Rose. Construction of experimental designs for mixed logit models allowing for correlation across choice observations. *Transportation Research Part B: Methodological*, 44 (6):720–734, 2010.

Michiel C. J. Bliemer and John M. Rose. Experimental design influences on stated choice outputs: An empirical study in air travel choice. *Transportation Research Part A: Policy and Practice*, 45(1):63–79, 2011.

Michiel C. J. Bliemer, John M. Rose, and Stephane Hess. Approximation of Bayesian efficiency in experimental choice designs. *Journal of Choice Modelling*, 1(1):98–126, 2008.

Sebastian Caussade, Juan de Dios Ortúzar, Luis I. Rizzi, and David A. Hensher. Assessing the influence of design dimensions on stated choice experiment estimates. *Transportation Research Part B: Methodological*, 39(7):621–640, 2005.

Keith Chrzan. Using partial profile choice experiments to handle large numbers of attributes. *International Journal of Market Research*, 52(6):827–840, 2010.

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

Christopher M. Gotwalt. Addendum to fast computation of designs robust to parameter uncertainty for nonlinear settings. *Technometrics*, 52(1):137–137, 2010.

Christopher M. Gotwalt, Bradley A. Jones, and David M. Steinberg. Fast computation of designs robust to parameter uncertainty for nonlinear settings. *Technometrics*, 51(1):88–95, 2009.

Ulrike Graßhoff, Heiko Großmann, Heinz Holling, and Rainer Schwabe. Optimal designs for main effects in linear paired comparison models. *Journal of Statistical Planning and Inference*, 126(1):361–376, 2004.

Heiko Großmann. Differences between analytic and algorithmic choice designs for pairs of partial profiles. In *mODa 10–Advances in Model-Oriented Design and Analysis*, pages 125–133. Springer, 2013.

Heiko Großmann, Heinz Holling, Ulrike Graßhoff, and Rainer Schwabe. Optimal designs for asymmetric linear paired comparisons with a profile strength constraint. *Metrika*, 64(1):109–119, 2006.

Heiko Großmann, Ulrike Graßhoff, and Rainer Schwabe. Approximate and exact optimal designs for paired comparisons of partial profiles when there are two groups of factors. *Journal of Statistical Planning and Inference*, 139(3):1171–1179, 2009.

Heiko Großmann, Ulrike Graßhoff, and Rainer Schwabe. A catalogue of designs for partial profiles in paired comparison experiments with three groups of factors. *Statistics*, forthcoming, 2014.

David A. Hensher. How do respondents process stated choice experiments? Attribute consideration under varying information load. *Journal of Applied Econometrics*, 21(6):861–878, 2006a.

David A. Hensher. Towards a practical method to establish comparable values of travel time savings from stated choice experiments with differing design dimensions. *Transportation Research Part A: Policy and Practice*, 40(10):829–840, 2006b.

David A. Hensher. Influence of vehicle occupancy on the valuation of car drivers travel time savings: Identifying important behavioural segments. *Transportation Research Part A: Policy and Practice*, 42(1): 67–76, 2008.

David A. Hensher and John M. Rose. Development of commuter and non-commuter mode choice models for the assessment of new public transport infrastructure projects: a case study. *Transportation Research Part A: Policy and Practice*, 41(5):428–443, 2007.

Joel Huber and Klaus Zwerina. The importance of utility balance in efficient choice designs. *Journal of Marketing Research*, 33(3):307–317, 1996.

John Douglas Hunt and John E. Abraham. Influences on bicycle use. *Transportation*, 34(4):453–470, 2007.

Roselinde Kessels, Bradley Jones, Peter Goos, and Martina Vandebroek. Recommendations on the use of Bayesian optimal designs for choice experiments. *Quality and Reliability Engineering International*, 24(6): 737–744, 2008.

Roselinde Kessels, Bradley Jones, Peter Goos, and Martina Vandebroek. An efficient algorithm for constructing Bayesian optimal choice designs. *Journal of Business & Economic Statistics*, 27(2):279–291, 2009.

Roselinde Kessels, Peter Goos, Bradley Jones, and Martina Vandebroek. Rejoinder: The usefulness of Bayesian optimal designs for discrete choice experiments. *Applied Stochastic Models in Business and Industry*, 27(3):197–203, 2011a.

Roselinde Kessels, Bradley Jones, and Peter Goos. Bayesian optimal designs for discrete choice experiments with partial profiles. *Journal of Choice Modelling*, 4(3):52–74, 2011b.

Roselinde Kessels, Bradley Jones, Peter Goos, and Martina Vandebroek. The usefulness of Bayesian optimal designs for discrete choice experiments. *Applied Stochastic Models in Business and Industry*, 27(3):173–188, 2011c.

Roselinde Kessels, Bradley Jones, and Peter Goos. An improved two-stage variance balance approach for constructing partial profile designs for discrete choice experiments. *Applied Stochastic Models in Business and Industry*, to appear, 2015.

Warren F. Kuhfeld and Randall D. Tobias. Large factorial designs for product engineering and marketing research applications. *Technometrics*, 47(2):132–141, 2005.

Franziska Kupfer, Roselinde Kessels, Peter Goos, Eddy Van de Voorde, and Ann Verhetsel. A discrete choice approach for analysing the airport choice for freighter operations in Europe. Technical Report 2013-28, Faculty of Applied Economics. University of Antwerp, 2013.

Becky P. Y. Loo, S. C. Wong, and Timothy D. Hau. Introducing alternative fuel vehicles in Hong Kong: views from the public light bus industry. *Transportation*, 33(6):605–619, 2006.

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

John Monahan and Alan Genz. Spherical-radial integration rules for Bayesian computation. *Journal of the American Statistical Association*, 92(438):664–674, 1997.

Ivan P. Mysovskikh. The approximation of multiple integrals by using interpolatory cubature formulae. In Ron de Vore and Karl Scherer, editors, *Quantitative Approximation*, pages 217–243. Academic Press, New York, 1980.

Gábor Nagy and Saïd Salhi. Location-routing: Issues, models and methods. *European Journal of Operational Research*, 177(2):649–672, 2007.

Wu Quan, John M. Rose, Andrew T. Collins, and Michiel C. J. Bliemer. A comparison of algorithms for generating efficient choice experiments. Technical Report ITLS-WP-11-19, Institute of Transport and Logistics Studies, University of Sydney, 2011.

John M. Rose and Michiel C. J. Bliemer. Constructing efficient stated choice experimental designs. *Transport Reviews*, 29(5):587–617, 2009.

Wafaa Saleh and Séona Farrell. Implications of congestion charging for departure time choice: work and non-work schedule flexibility. *Transportation Research Part A: Policy and Practice*, 39(7):773–791, 2005.

Said Salhi and Graham K. Rand. The effect of ignoring routes when locating depots. *European Journal of Operational Research*, 39(2):150–156, 1989.

Zsolt Sándor and Michel Wedel. Designing conjoint choice experiments using managers' prior beliefs. *Journal of Marketing Research*, pages 430–444, 2001.

Patrick Schittekat, Joris Kinable, Kenneth Sörensen, Marc Sevaux, Frits Spieksma, and Johan Springael. A metaheuristic for the school bus routing problem with bus stop selection. *European Journal of Operational Research*, 229(2):518–528, 2013.

Ipek N. Sener, Naveen Eluru, and Chandra R. Bhat. An analysis of bicycle route choice preferences in Texas, US. *Transportation*, 36(5):511–539, 2009.

Nebiyou Y. Tilahun, David M. Levinson, and Kevin J. Krizek. Trails, lanes, or traffic: Valuing bicycle facilities with an adaptive stated preference survey. *Transportation Research Part A: Policy and Practice*, 41(4):287–301, 2007.

Ann Verhetsel, Roselinde Kessels, Peter Goos, Toon Zijlstra, Nele Blomme, and Jeroen Cant. Location of logistics companies: a stated preference study to disentangle the impact of accessibility. *Journal of Transport Geography*, 42(1):110 – 121, 2015.

Thibaut Vidal, Teodor Gabriel Crainic, Michel Gendreau, and Christian Prins. Heuristics for multi-attribute vehicle routing problems: a survey and synthesis. *European Journal of Operational Research*, 231(1):1–21, 2013.

Baojin Wang, David A. Hensher, and Tu Ton. Safety in the road environment: a driver behavioural response perspective. *Transportation*, 29(3):253–270, 2002.

Jie Yu, Peter Goos, and Martina Vandebroek. Comparing different sampling schemes for approximating the integrals involved in the efficient design of stated choice experiments. *Transportation Research Part B: Methodological*, 44(10):1268–1289, 2010.