New algorithms for generating balanced efficient stated preference designs

Collins Teye-Ali, Peter Davidson, Rob Culley
Peter Davidson Consultancy, Ravens Lane, Berkhamsted, HP4 2DX, UK

Abstract

In transport we need to understand and quantify the importance people attach to the different attributes of travel and how changes in these attributes might affect what they do. The traditional approach was to use an orthogonal fractional factorial design but this places severe limitations on the number of variables and their levels as the number of choice situations generally increases exponentially with increasing number of attributes. This led to the recent design approach based on the principle of '*efficiency*' rather than the traditional principle of 'orthogonality'. Efficient designs are typically nonorthogonal; however they are efficient in the sense that the variances and covariances of the parameter estimates are minimized. However, existing algorithms such as modified Fedorov or RSC-algorithm for generating such designs do not guarantee that the optimized design is level balanced, especially if the starting design is unbalanced. A balanced design is desirable for least two reasons; In the presence of an unbalanced design, psychological side effects are likely to determine biased results since it is very likely that a respondent could think that there is a reason why a level appears more often than others in a presented set of choice situations; Having attribute level balance ensures that the parameters can be estimated well on the whole range of levels, instead of just having data points at only one or a few of the attribute levels, which intuitively provides a good basis for estimation.

We have proposed new algorithms for generating balanced efficient designs. Experimental analyses show that designs generated by our proposed algorithms are in additional to being balanced, statistically more efficient than the modified Fedorov algorithm.

1. Introduction

In transport we need to understand and quantify the importance people attach to the different attributes of travel and how changes in these attributes might affect what they do. This analysis was traditionally based on Revealed Preference (RP) data; that is choices and decisions that have actually been made in the marketplace and the attributes that are measurable or quantifiable (Train, 2009). But there are inherent practical limitations associated with the RP approach, largely connected with survey costs and lack of variations in relevant attributes during estimation (Train, 2009). Also in many practical applications the Analyst will want to examine people's responses in situations that do not currently exist, such as the demand for park and ride and revealed-preference data are simply not available for these new situations. Even for choice situations that currently exist, there may be insufficient variation in relevant attributes to allow estimation with revealed-preference data. This lack of variation could lead to attributes that are most important to travellers exhibiting the least variation leading to insignificant coefficients as their importance might be difficult to detect (Train, 2009).

Stated preference (SP) is aimed at solving the problems encountered in using revealed preference data (Hess and Rose, 2008; Sanko et al, 2002; Louviere *et al.*, 1991; Toner, et al, 1999; Toner et al, 1998; Hensher et al, 1996; Hensher, 1994; Fowkes and Wardman, 1993). In a typical SP experiment respondents are presented with hypothetical choice situations that are described with a set of well defined attributes (e.g., travel time, cost, etc), describing alternatives, where each attribute may have two or more levels (Toner et al, 1998). The hypothetical alternatives and the attributes describing them are generated through experimental design (Kessels *et al.,* 2006; Sándor and Wedel 2001, 2002, 2005). A simple and well known experimental design is the *Full factorial design.* A full factorial design consists of all possible combinations of the levels of each attribute (Rose and Bliemer, 2007). For example, with 5 attributes, each with 3 levels, there are 243 ($=3^5$) possible different choice situations. The attractiveness of this type of design is the fact that it has many desirable properties including being able to estimate all main effects, two-way, and higher-order interactions from the resulting data. This means the Analyst can obtain the maximum possible information about the design parameters through this type of experiment. The disadvantage with this experiment is the practical difficulty of presenting all the 243 choice situations to each respondent. Additionally many of the choice situations do not add any significant information about the design parameters. This makes this type of design too cost-prohibitive, tedious and unattractive in practice (Rose, and Bliemer, 2007).

For this reason, many Analysts often use a fraction of the 'full factorial design' called a *fractional-factorial design*, which still retains most of the desirable properties of the full-factorial design. The basic difficulty is how to construct such

a fractional-factorial design that can retain as many desirable properties as possible of a full factorial design but with much fewer choice situations.

A special type of fractional-factorial design is the *orthogonal array*, in which all estimable effects are uncorrelated. Orthogonal arrays are categorized by their resolution (Kuhfeld 2005), where the resolution identifies which effects, possibly including interactions, are estimable. Higher resolutions require larger choice situations. Orthogonal arrays come in specific numbers of choice situations (e.g., 16, 18, 20, 24, 27, 28, etc) for specific numbers of attributes with specific numbers of levels. Although these designs are orthogonal and significantly reduce the number of choice situations compared with full factorial designs, they can only be available for relatively small number of very specific problems for the following reasons:

1. The number of choice situations also generally increases exponentially with increasing number of attributes.

2. The number of choice situations cannot be freely chosen by the Analyst (Bliemer and Rose, 2007; Kuzmanovia and Vukmirovia, 2005)

3. They may not be available when the number of attribute levels is different for most of the attributes and when some combinations of attribute levels are unfeasible (Kuzmanovia and Vukmirovia, 2005),

4. They may also contain irrelevant choice situations as in full factorial designs (Bliemer and Rose, 2007)

5. Generally two-way interactions and higher-order interactions are not estimable (Bliemer and Rose, 2007).

The above limitations have led to the current state-of-the-art design principles based on design '*efficiency*' rather than the traditional principle of orthogonality (e.g., Huber and Zwerina 1996; Kessels *et al.* 2006; Sándor and Wedel 2001, 2002, 2005). Efficient designs aim to produce stable and reliable parameter estimates in a fractional design setting by minimising at least one property of the information matrix (e,.g, the determinant or trace) of the log-likelihood function of the chosen logit model (Huber and Zwerina, 1996). In this paper we focussed on minimizing the determinant (D-error) of the information matrix of the multinomial logit (MNL) models. The choice of multinomial logit model in this study was partly motivated by the fact that it is probably the most popular discrete choice model for designing stated choice experiments and as noted by (Ortuzer and Willumsen, 2011) designs optimised for the MNL model typically perform well when analysed using other model forms such as Nested logit (NL) or Mixed logit models (MXL) models (Bliermer et al, 2009; Sandor and Wedel, 2002; Bliermer and Rose, 2008). Although there exist several other measures of efficiency such as the A-error (which minimizes the trace of the information matrix), G-error

(which minimizes the maximum prediction variance over the design region) and the V-error (which minimizes the average prediction variance over the design region), it was demonstrated in Kessels et al, (2006) that the D-error measure of efficiency is the best criterion for measuring design efficiency and is also computationally very efficient.

In addition to the D-error or D-efficiency, there are four other desirable properties in the literature that characterise an efficient design as noted in Huber and Zwerina (1996). These desirable properties are: *level balance* (attribute levels are presented in equal frequency in each choice set); *orthogonality* (attribute levels are uncorrelated to one another and their combinations do not exhibit a certain pattern); *minimum overlap* (for each choice set the repetition of an attribute level is minimised); *utility balance* (alternatives within a choice set should have nearly equal attractiveness to the respondents)..

Kessels et al, (2006) introduced a measure scaled to be in the range 0 and 100 for computing the *utility balance* of a given design with values close to 100% considered more utility balanced. In this paper we proposed a measured also scaled between 0 and 100 for measuring the level balance of an efficient design.

Finding an efficient design usually involves selecting the required or desired number of choice situations from a candidate set (list of potential choices or scenarios) that is usually generated by full or fractional factorial designs (Rose and Bliemer 2007). The problem that rises is how to find an algorithm that *efficiently* selects the best or optimal desired choice situations from the candidate set. An efficient algorithm is an algorithm that finds the optimum solution of every instance of the problem in polynomial time. Unfortunately no such algorithm exists for this type of problem (Garey and Johnson 1979). Existing algorithms such as branch and bound, dynamic programming and 'brute force' are known to have exponential running times. For example a design with only 5 attributes each with 3 levels will result in 243 possible combinations using full factorial design. Assume the analyst wants to select the best 12 from the 243 possible lists using the 'brute force' approach. This algorithm will find the best solution by evaluating over 6.7×10^{19} possible choice situations and select the best. That means if it takes a computer a microsecond (10^{-6}) to evaluate one design, then this algorithm will take about 2 million years to select the best design

The fruitful approach is to relax the notion of global optimality and be content with algorithms that generate near-optimal designs with polynomial running times. These algorithms are generally called heuristic algorithms as there is no way of determining exactly how far they are from the 'true' solution although they are known to produce good solutions. Special heuristic algorithms developed for efficient designs include Dykstra's (1971) sequential search method, which starts with an empty design and adds candidate points so that the chosen efficiency criterion is maximized at each step. This algorithm is fast, but it is not very

reliable in finding a globally optimal design. Also, it always finds the same design (due to a lack of randomness).

The Federov (1972) algorithm simultaneously adds one candidate point and deletes one design point. Cook and Nachtsheim (1980) define a modified Federov algorithm that finds the best candidate point to switch with each design point from the candidate points. The resulting procedure is generally as efficient as the Federov algorithm in finding the optimal design and it is up to twice as fast. For example consider a desired design with $n = 5$ choice situations and a candidate set of $N = 20$ choice situations. The Fedorov algorithm calculates 100 D-error values for all the $n * N = 100$ possible couples and suggests only a couple for exchange. By contrast, the modified version of the algorithm starts with the first design point and compares it with the 20 candidate points in the candidate set and suggests the candidate point to swap with the first design point. By doing so the Modified Fedorov algorithm calculates only 20 D-error values for an exchange to be made. The algorithm then goes on with the next design point and so on.

The RSC-algorithm comprises three sub-algorithms, namely relabelling (R), swapping (S) and cycling (C), thereby constructing a so-called RSC-design. Relabelling and swapping have been introduced by Huber and Zwerina (1996) in combination with the use of nonzero prior point coefficients. Cycling has been added by Sandor and Wedel (2001). The Relabelling modifies a design by permuting the levels of the attributes across choice sets and searches for a combination of permutations for which the corresponding design has the highest efficiency. Swapping involves switching two attribute levels among alternatives within a choice set and testing if the swap improves the criterion value. Cycling is a combination of cyclically rotating the levels of an attribute and swapping them. All cycles and swaps are examined for design improvements and if an improvement emerges, the corresponding design is stored. If, after a while, no improvement is possible, then the last stored design is the one with the largest efficiency gains and is called the optimal RSC-design.

2.0. Proposed algorithms for generating efficient designs

2.1. Level Balance efficiency criterion:

We proposed a criterion called *level balanced efficiency (LBeff)* scaled between 0 and 100% as an indicator of how balanced a given design is. First the level balance for each alternative is computed followed by the weighted average over all the alternatives.

The level balance (LB) for alternative *j* is given as:

$$LB_j = 100 * \left(\frac{1}{A_j} \sum_{i=1}^{A_j} \eta_{ij,} \right)$$

Where

$$\eta_{ij} = Min \left\{ \frac{f_{lij;}}{S / L_{ij}}, \forall l \right\}$$

The overall level balance over all alternatives:

$$LBeff = \frac{\sum_{j=1}^{J} A_j * LB_j}{\sum_{j=1}^{J} A_j}$$

Where f_{lij} is the number of occurrences of level *l* of attribute *i* in alternative *j*
L_{ij} is the number of levels of attributes *i* in alternative *j*
S is the number of choice scenarios and
A_j is the number of attributes available for alternative *j*.

By construction, a design with say *LBeff* of 60% means that the design is on average 60% balanced. For example, given the design below with two alternatives and two attributes each with two levels:

Scenario	Alternative	X1	X2
1	1	0	1
1	2	0	0
2	1	1	0
2	2	0	1
3	1	0	1
3	2	0	1

4	1	1	0
4	2	1	0
5	1	0	0
5	2	0	1
6	1	1	1
6	2	1	0
7	1	0	1
7	2	1	1
8	1	1	1
8	2	0	1

Separating the design into alternatives and counting the frequency of each attribute level we have:

Alternative 1

Levels	X1	X2
0	4	3
1	4	5
eta	1	0.75

Alternative 2

Levels	X1	X2
0	5	4
1	3	5
eta	0.75	0.75

The level balance for alternative 1 = (1+0.75)/2 = 0.875
The level balance for alternative 2 = (0.75+0.75)/2 = 0.75
The overall level balance over the two alternatives = (2*0.875 + 2*0.75)/(2+2) = 0.8125

Thus the above design is 81.25% balanced.

2.2. PDC Algorithm

The basic idea behind the algorithm we proposed is to randomly select a design from a candidate set, ensure it is level balanced and then employ an improvement algorithm (e.g., swap or simulated annealing algorithm) to improve the solution.

The algorithm has the following features:

1. Generate a candidate set using a full or fractional factorial design.
2. Randomly select the desired number of choice situations (*Xcur*) from the candidate set
3. Use the PDC level balance algorithm (Section 2.3) to balance the design
4. Apply the swap or Simulated annealing algorithm (Section 2.4) to optimize the design
5. Repeat steps 2 to 4 until the specified number of iterations or a given stopping criterion is reached.

2.3 PDC Level balanced algorithm

1. For each attribute under each alternative in the design compute the maximum number of times that each level for that attribute can occur

   $$L_{\max} = \left\lceil \frac{S}{L} \right\rceil$$ (The smallest positive integer greater than or equal to *S/L*)

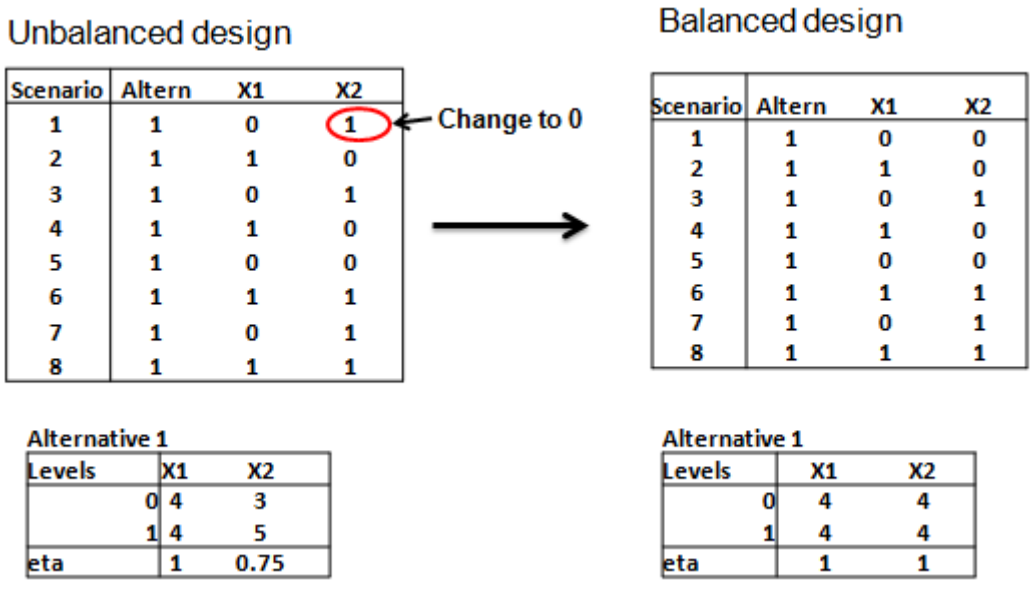
   Where *L* is the number of levels of current attribute, *S* the number of choice situations

   1.1. Count the frequency of each level
   1.2. Find the level that occurred least *(fmin)* and the most *(fmax)*
   1.3. If *fmax < Lmax* goto step 2 else compute *fmax – Lmax*
   1.4. Replace *fmax-Lmax* number of the level that occurred the most with the level that occurred the least.
   1.5. Repeat steps 1.1 to 1.5 until no level is greater than *Lmax*.

2. Repeat step 1 for all attributes for that alternative

3. Repeat steps 1 and 2 for all alternatives

For example figure 2 below shows how an unbalanced design is balanced.



Scenario	Altern	X1	X2
1	1	0	1
2	1	1	0
3	1	0	1
4	1	1	0
5	1	0	0
6	1	1	1
7	1	0	1
8	1	1	1

Scenario	Altern	X1	X2
1	1	0	0
2	1	1	0
3	1	0	1
4	1	1	0
5	1	0	0
6	1	1	1
7	1	0	1
8	1	1	1

Levels	X1	X2
0	4	3
1	4	5
eta	1	0.75

Levels	X1	X2
0	4	4
1	4	4
eta	1	1

Fig 2: Balancing an unbalanced design

2.4. Improvement algorithms

We proposed two improvement algorithms for improving the balanced design. The two algorithms are the swap and the simulated annealing algorithms. Each of these two algorithms is described detail below.

2.3.1 The Swap algorithm

The basic principle behind the swap algorithm is to look for the best position of each level of an attribute of the design. The algorithm starts with the attribute level in first position, and suggests swapping it with the level at the second position provided the couple are not the same. If the swapping improves the design, the swap is made permanent, and this new level in the first position then competes with the level in third position and so on. A swap is only made permanent if it results in improved design efficiency; otherwise the two levels remain in their original positions. The algorithm starts with the first attribute under the first alternative and then loops over all attributes and then over all alternatives.

2.3.2. Simulated Annealing version of the swap algorithm

Simulated Annealing (SA) is a random-search optimisation technique (Kirkpatrick et al, 1983) inspired by the manner in which a metal crystallizes in the process of annealing or in which liquids freeze. The technique involves heating and controlled cooling of a material to increase the size of its crystals and reduce their defects. The heat causes the atoms of the metal to become unstuck from their initial positions (a local minimum of the internal energy) and wander randomly through states of higher energy; the slow cooling gives them more chances of finding configurations with lower internal energy than the initial one.
The law of thermodynamics state that at temperature, T, the probability of an increase in energy of magnitude ΔE is given by:

$$\exp\left(-\frac{\Delta E}{L*T}\right)$$

Where T is the temperature, L is known as Boltzmann's constant. This constant is usually dropped as it was introduced to cope with different materials.

By analogy with this physical process, each step of the SA algorithm attempts to replace the current solution by a random solution (often constructed from solutions near the current solution). The new solution may then be accepted with a probability given above. This probability depends both on the difference between the corresponding function values and also on a global parameter T (called the *temperature*), that is gradually decreased during the process. The dependency is such that the choice between the previous and current solution is

almost random when *T* is large, but increasingly selects the better or "downhill" solution (for a minimization problem) as *T* goes to zero. The allowance for "uphill" moves potentially saves the method from becoming stuck at local optimum—which are the bane of greedy algorithms such as the 'swap' algorithm.

In this paper we adapted this optimization technique to generate efficient designs. The version of the algorithm we implemented is described as follows:

Step1. Initialisation

1.1. Let *Xcur* be the current design with D-error *D(Xcur)* and *Xnew* with D-error *D(Xnew)* be the design after a swap operation on *Xcur. X** is the optimum design during the iterations. Let *Lm, Ln (m,n =1, 2, ..S)* be the level points of an attribute in choice situation *s (s = 1,…, S)*.

1.2. Let *Xcur = Xnew; X* = Xnew*. Compute and set *D(Xnew) = D(Xcur) = D(X*).* Choose *Initial temp T= To* and a cooling schedule *b*

Step2. Apply the following procedure for each attribute in the design

2.1. Generate design *Xnew* by swapping level points *Lm* and *Ln* in design *Xcur* using the swapping technique in section 2.3.1.

2.2. Compute the D-error, *D(Xnew)* for the new design.

2.3. If *D*(*Xnew*) < *D*(*Xcur*),

2.3.1 *Xcur =Xnew* , *D(Xcur) = D(Xnew)*

2.3.2. if *D(Xnew)<D(X*); D(X*) = D(Xnew) and X*=Xnew*

2.4. Else generate a uniform random number *U Uniform (0,1)*

2.4.1. If

$$U < \exp\left(\frac{-d}{T}\right)$$

Xcur =Xnew, *D(Xcur) = D(Xnew)*

Where d = D(*Xnew*) - *D*(*Xcur*)

2.4.2. Else

Discard the new design Xnew.

2.4.3. Reduce the system temperature according to the cooling schedule.

2.5. Repeat steps 2.1 to 2.4 for all *Lm* and *Lm* where *Lm* is not equal to *Ln*

Step3. Next attribute in the design

3.1. Xcur = X*

3.2. Repeat step 2 for attributes.

Step4. Next alternative in the design

4.1. Repeat steps 2 and 3 for all alternatives and stop.

The choice of the initial temperature and the temperature Schedule (which determines the temperature to use for the current iteration step) has a significant impact on the method's effectiveness. Unfortunately, there are no choices of these parameters that will be good for all problems, and there is no general way to find the best choices for a given problem.

In this study we experimentally chose the initial temperature to be 1 and then reduced by 10% in each iteration. Thus

$T = To*0.9$

Where T is the updated temperature, To is the initial temperature.

3. Experimental study of algorithms for generating efficient designs

3.1. Introduction

For the sake of identification we called the PDC algorithm with swap improvement algorithm the *PDC algorithm*, whilst the simulated annealing version is called the *PDC+SA algorithm*. A simple experiment was conducted to ascertain the strengths and weaknesses of the two proposed heuristics and the modified Fedorov (MFD) algorithm.

All the algorithms have a polynomial running time, and so we concentrated on the quality of the designs. Conducting experiments to test the quality of the heuristic algorithms is more scientific (Rardin and Uzsoy 2001) and also gives an indication of how much the results can be trusted. Thus analysis of using averages (measure of central tendencies) and standard deviations (measure of variability) for a given set of instances, gives a better picture of how the algorithms may perform in practice.

3.2. Assumptions about design parameters

In the literature there exist least three types assumptions about the about the design parameters. In the first approach, the Analyst assumes all the parameters are zero (e.g., Burgess and Street 2003; Huber and Zwerina 1996; Street and Burgess 2004; Street *et al.* 2001). This approach is probably the most common in practice as the non-linear model becomes linear and so allows the use of linear techniques such as orthogonal coding to generate efficient designs. As noted in Kuhfeld et al, (1994) efficient designs for linear models such as linear regression do not depend on the actual values of the design parameters. Thus the Analyst has the freedom of generating the design using the multinomial logit model, but can analyse the data using any of the complex model types (e.g., Multinomial logit (MNL), Nested logit (NL), Cross nested logit (CNL), mixed logit (MXL), etc). Lazari and Anderson (1994) and Kuhfeld et al. (1994), argue that efficient designs for linear models also work well for the nonlinear choice models. One limitation of this design type is that fact that the true parameter values will in reality be non-zero and so the design could be a sub-optimal design (Hess et al, 2008).

The second approach assumes non-zero design parameter values (e.g., Huber and Zwerina 1996; Rose and Bliemer 2005). The values are usually obtained from the literature, experience or previous studies. The designs resulting from this type can best be described as local optimum as different values may result in different designs. Additionally, not all parameters values can be borrowed and used in the design, especially parameters associated with qualitative attributes. Another strong limitation of this type of design is that it is model specific. As stated, earlier efficient designs are obtained by minimizing an objective function that is a function of the covariance matrix of an assumed model. That is the analyst must have a priori assumption about the model to fit the data on, so that the covariance matrix of the model is minimised during the design stage. In practice it will be a big gamble to conduct the survey with the assumption of fitting it on only one model type because there is every possibility that the model may not result in reliable or intuitive parameter estimates. Although it is generally accepted to use a design based on one discrete choice model as an approximation for other choice models, there is no way of determining how far it is from optimality.

The third approach is called the Bayesian approach (Sándor and Wedel ,2001). This type of design does not assume a precise knowledge of the design parameter values but assumes that the true parameter values fall within some distribution of possible priors. The designs are created by optimising over several parameter values drawn from the chosen probability distribution. This design type inherits all the limitations of the second approach except that the design obtained is averaged over several values. Another limitation about the type of design is the choice of the distribution of each design parameter value(s) and how many draws

should be used before the Bayesian measure of efficiency will converge to the true efficiency level.

In this study we adopt the first approach, where all the parameter values are assumed to be simultaneously zero. Our choice of this approach was partly motivated by the following reasons:

1. It allows us to generate hundreds of different design instances and compare the performance of different algorithms without worrying about the values of the design parameters.
2. It makes it possible to compute a design efficiency measure called *D-efficiency* to range from 0 and 100% thus providing useful insights into the meaning of 100% efficiency and less than 100% efficient designs.
3. Freedom to minimise the covariance matrix of a multinomial logit (MNL) model in generating the designs (McFaddden 1974). With the assumption of zero parameter values there is no benefit in using a more complex model as the unconditional probabilities of the alternatives in each choice set of all discrete choice models (e.g., NL, GNL, MXL etc) reduce to those of the multinomial logit model.
4. The *utility balance* (Kuhfeld et al, 2004) property of efficient designs is automatically achieved as all alternatives have equal probability and hence equal chance of being chosen. The importance of utility balance designs is demonstrated in Huber, and Zwerina, (1996).

3.3. Instance Generation

Since there is essentially no reference benchmark available, we developed our own instance generator for this study. In this experiment, 3 different attribute levels are drawn from the set {2,3,4}, 3 different alternatives numbers selected from the set {2,3,4}, 9 different attribute numbers selected from the set {2,3,4,5,6,7,8,9,10}, and 5 different choice situations selected as a function of the number attributes and levels such that a feasible and level balance design, can be achieved by the algorithms. That is the minimum number of choice situations selected was the smallest multiple of the number of attribute levels greater than or equal to the number of attributes. For example if there are 4 attributes, each with 3 levels, then the smallest choice situation becomes 6. The next one will be 6 +3 = 9, followed by 6+2*3 = 12 and so on. We also restricted the maximum number of choice situations to be less than L^A, where L is the number of levels, and A is the number of design attributes since it is the number of choice situations in a corresponding full factorial design. So in all we generated a total of 371 feasible design instances for each of the three algorithms.

3.4. Analysis of results

In this section, we compare the performances of our two proposed algorithms and the well known Modified Fedorov algorithm (MFA) in terms of D-efficiency and level balance efficiency. Both performance criteria are scaled in the range 0 and 100 such that designs with D-efficiency or level balance efficiency values approaching 100% are considered more desirable or more efficient and balanced. The algorithms proposed are the PDC and PDC+SA algorithms. The PDC+SA algorithm was proposed to improve the performance of the PDC algorithm so that it doesn't get trapped in local optimum during the swapping process.

3.4.1 Analysis of results with Level balance efficiency

The main strength of the proposed PDC algorithms is the guarantee of generating level balance designs irrespective of the number of attributes, attribute levels, alternatives or number of choice situations. The ability of the modified Fedorov to generate level balance designs worsened with increasing number of attribute levels as shown figure 3. The two PDC algorithms produced level balanced designs for every instance of the problem. The Modified Fedorov algorithm produced less balanced designs with increasing attribute levels. On average, the modified Fedorov algorithm generated designs that are only 67% balanced. The PDC algorithms on the other hand were 100% level balanced in all cases. Constraining the Modified algorithm to produce level balance designs could most likely produce worse efficiency results.

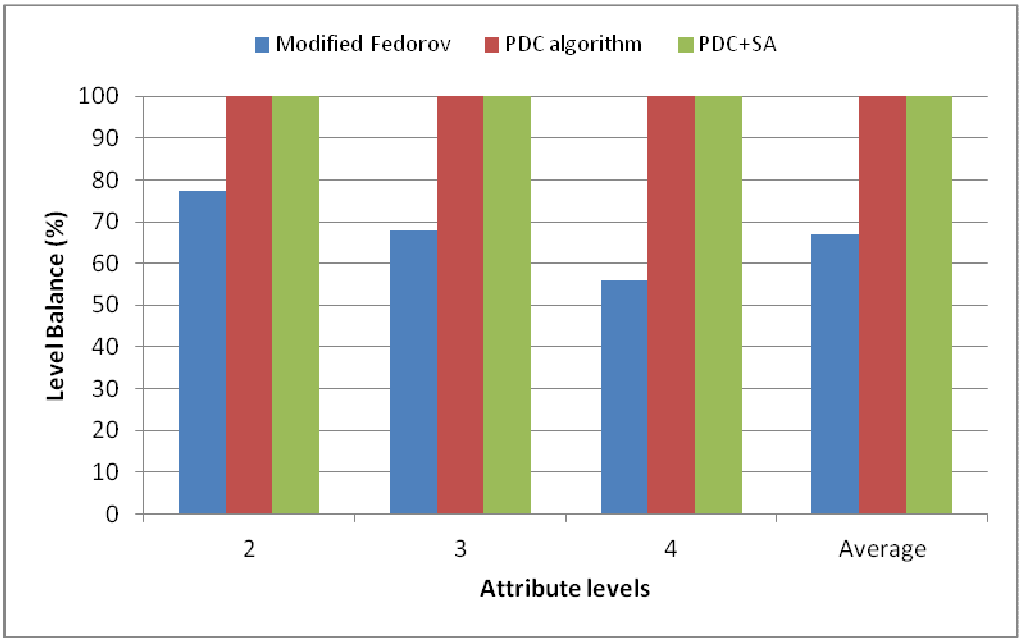


Fig 3: Level balance of the three algorithms averaged over all the 371 instances

3.4.2 Analysis of results with D-efficiency

We first investigated the performances of the algorithms by fixing the other variables but varying the alternatives. The result for this analysis is shown in figure 4. The figure shows that all the algorithms increase in efficiency with increasing number of alternatives. The two PDC algorithms are consistently better than the modified Fedorov algorithm over all alternative numbers. There is however no noticeable difference between the 2 PDC algorithms.

We also investigated the performances of the algorithms by fixing the other variables but varying the attribute levels. Figure 5 shows the results for this analysis. The figure shows that the performances of the algorithms got worse with increasing attribute levels.
The two PDC algorithms were again consistently better than the modified Fedorov algorithm over all attributes levels. The figure also reveals that the reduction in performance was worse for the modified Fedorov algorithm with increasing attribute levels than the 2 PDC algorithms. Here again, we found no noticeable difference between the 2 PDC algorithms.

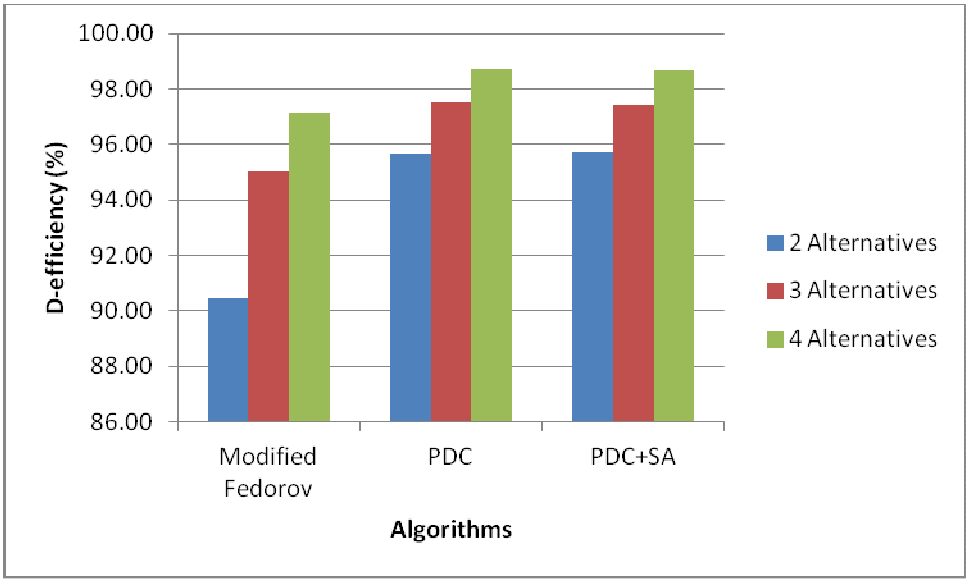


Fig 4: Algorithms performaces with increasing number of alternatives

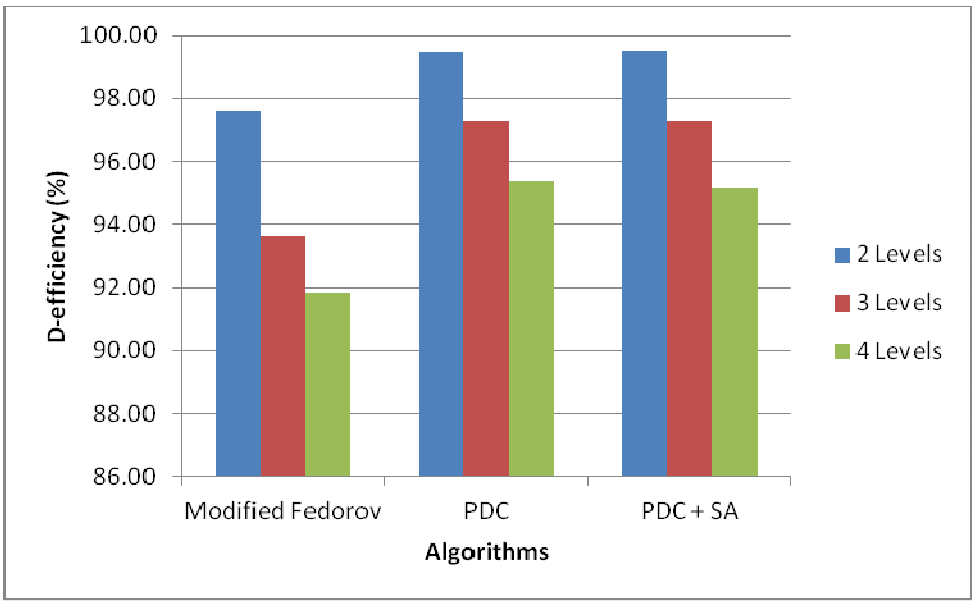


Fig 5: Algorithms performaces with increasing number of attribute levels

3.4.3. Overall performance of the algorithms

Here we look at the overall performances of the three algorithms over the 371 generated design instances for each algorithm. We also conducted hypothesis tests to determine whether the noticeable differences in the performances of the algorithms were significant.

Our analysis of the results showed that the average *D-efficiency* of the modified Fedorov (MFD) algorithm (94.28%) could be increased to 97.33% by using the PDC algorithm and to 97.28% by using the PDC+SA algorithm. Thus on average the PDC and the PDC+SA algorithms were 3.05% and 3.0% respectively better than the modified Fedorov algorithm as shown in table 1 and figure 6. The higher standard deviation of the modified Fedorov algorithm also implied that the Modified Fedorov algorithm was more likely to generate worse designs than the two PDC algorithms. Between the two PDC algorithms, the simulated annealing algorithm performed slightly better than swap version under two level attributes. The swap algorithm was however slightly better than the simulated annealing under the 4 level attributes.

Finally we conducted a hypothesis test to determine if the noticeable differences in the performances of the algorithms were significant. The analysed results are shown in table 2 and 3. The *p-value* in table 2 was approximately 0 indicating the performances of the three algorithms were significantly different at 95% confidence level. This conclusion was supported by further hypothesis testing using Fisher's least significant difference (LSD) test. Thus the performances of the two PDC algorithms were shown to be significantly better than the modified

Fedorov algorithm. There was, however no difference in efficiency between the two PDC algorithms. The LSD test results are shown in table 3.

Table 1: Average D-efficiencies with standard deviations for the generated design instances

	Mean			Standard Deviation		
Levels	Modified Fedorov	PDC	PDC + SA	Modified Fedorov	PDC	PDC + SA
2	97.59	99.49	99.54	3.82	0.76	0.70
3	93.63	97.28	97.27	6.08	3.85	3.66
4	91.85	95.36	95.19	6.82	5.27	5.56
Average	94.28	97.33	97.28	6.21	4.18	4.27

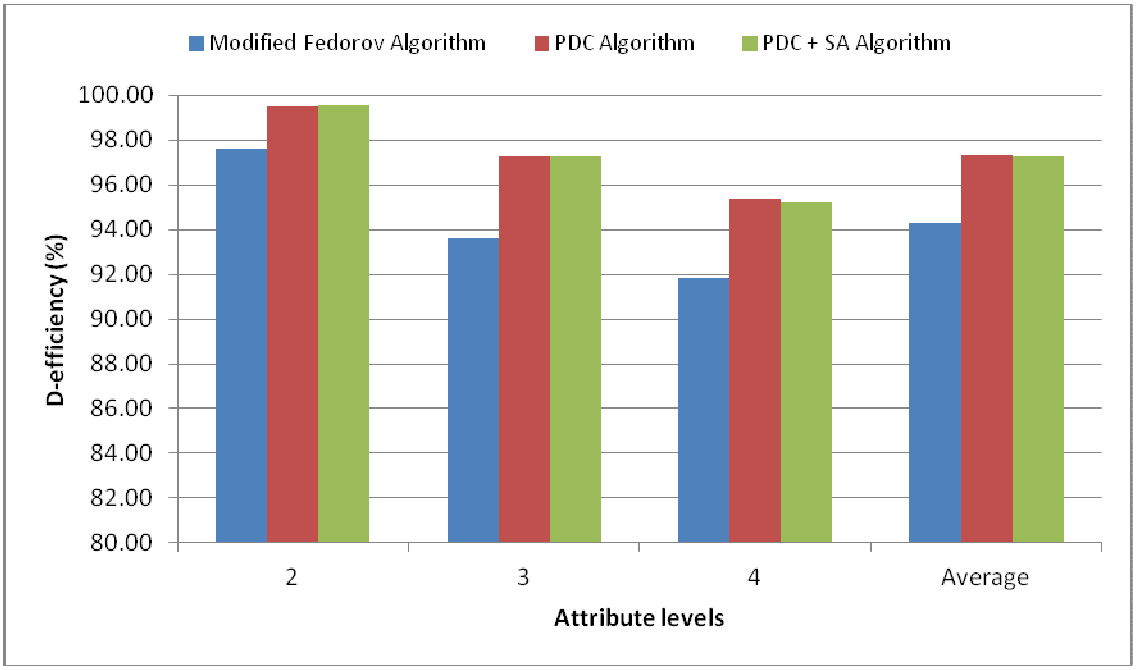


Fig 6: Overall performances of the three algorithms

Table 2: Analysis of Variance (ANOVA) of experimental results

Source of Variation	*SS*	*df*	*MS*	*F*	*P-value*	*F crit*
Between Groups	3204.10	2	1602.05	63.02	1.17E-26	3.0038
Within Groups	28369.45	1116	25.42			
Total	31573.54	1118				

Table 3: Fisher's least Significant difference (LSD) hypothesis results

Algorithms	Difference	Fisher LSD	Conclusion
MFD vs PDC	3.5731	0.7244	Reject
MFD vs PDC+SA	3.6058	0.7244	Reject
PDC vs PDC+SA	0.0328	0.7244	Do not reject

4. Conclusion

The objective of this paper was to attempt to develop the best possible heuristic algorithms that are capable of generating level balance designs without significantly affecting the efficiency of the design. Two algorithms were proposed and their performances were compared with the well known Modified Fedorov algorithm for generating efficient designs. The proposed heuristics were called PDC and PDC+SA algorithms and the qualities of designs they generated have been tested through experimental studies.

The experimental studies revealed that the ability of the modified Fedorov to generate level balance designs worsened with increasing number of attribute levels. The overall average level balance efficiency was only 67%. The PDC algorithms on the other hand were 100% level balanced in all cases. In terms of D-efficiency, PDC and PDC+SA algorithms are on average 3.05% and 3.0% respectively better than the modified Fedorov algorithm

The PDC+SA algorithm was proposed to improve the performance of the PDC algorithm so that it did not get trapped in local optima during the swapping process. However the PDC+SA ability to escape from the local optimum heavily depended on the choice of initial temperature and the cooling schedule, which seemed to be dependent of the problem instance. More research is needed to test for suitable temperatures and cooling schedules, preferable those that can be generated for a given problem instance. Although there were few cases where the PDC+SA outperformed the PDC especially with lower attribute levels, the average performance of the two algorithms were not significantly different. More work is also needed to test the algorithms on real problems with possible prior or Bayesian parameter assumptions.

References

Axhausen, K.W. and Polak, J.W. (1990) `Choosing the type of parking: stated preference experiments in the UK and Germany', *Working Paper*, **530**, Transport Studies Unit, Oxford University.

Axhausen, K.W. and Polak, J.W. (1991) "Choice of parking: stated preference approach" Transportation 18 **59-81**

Barone, S., Lombardo, A. (2004), "Service Quality Design through a Smart Use of Conjoint Analysis", *The Asian Journal on Quality*, vol. 5, n. 1, pp. 34-42.

Barone, S., Lombardo, A. (2005a), "Balanced Asymmetrical Nearly Orthogonal Designs for First and Second Order Effect Estimation". To appear in *Journal of Applied Statistics*

Barone, S., Lombardo, A. (2005b), "Construction and Optimality of a Special Class of Balanced Designs".

Bhat C., Koppelman, F.S., (2006): A self instructing course in model choice modelling: Multinomial and Nested logit models. Prepared for the US Department of Federal Transit Administration (http://www.caee.utexas.edu/prof/bhat/home.html)

Ben-Akiva, M. & Lerman, S.R. (1985) *Discrete Choice Analysis: Theory and Application to Travel Demand,* MIT Press, Cambridge

Bliemer, M.C. and Rose, J.M. (2009) Efficiency and Sample Size Requirements for Stated Choice Experiments, *Transportation Research Board Annual Meeting*, Washington DC January.

Bradley, M.A. & Daly, A.J. (1997) Estimation of logit choice models using mixed stated preference and revealed preference information. In Stopher P. & Lee-Gosselin, M. (Eds.) *Understanding Travel Behaviour in an Era of Change.* Elsevier, Oxford

Bradley, M.A. & Daly, A.J. (1992) Use of the logit scaling approach in stated preference analysis. Paper presented at the 7th World Conference on Transport Research, Lyon, July.

Bradley, M.A. & Daly, A.J. (1992) Stated preference surveys. In:Ampt ES, Richardson AJ & Meyburg AH (eds), Selected Readings in Transport Survey Methodolody. Eucalyptus Press, Melbourne (pp 31-35)

Burgess, L., Street, D.J. (2005), "Optimal designs for choice experiments with asymmetric attributes", *Journal of Statistical Planning and Inference*, in press.

Cook, R. D. and Nachtsheim, C. J. (1980). A comparison of algorithms for constructing exact *D*-optimal designs, *Technometrics* 22: 315{324. Fedorov, V. V. (1972). *Theory of optimal experiments*, New York: Academic Press.

Dykstra, Otto (1971): The Augmentation of experimental data to maximize $(X'X)^{-1}$,Technometrics, 13 (August), 682-88

Fedorov, Valery (1972): Theory of optimal experiments, translated and edited by W.J.. Studden and E.M., Klimko. New York: Academic press.

Fowkes and Wardman (1993): Non-orthogonal stated preference design, PTRC SAM
Gary M., and Johnson D.S. (1979): Computers and Intractability: A guide to the theory of NP-Completeness. Bell Laboratories.

Geweke, J. (1996). Monte Carlo simulation and numerical integration, in *Handbook of Computational Economics*, Vol. 1, Ch. 15, Amman H. M., Kendrick D. A., Rust J.,eds. Amsterdam: Elsevier Science, 731{800.

Hensher, D.A. (1994) Stated Preference Analysis of choices: the State of Practice, *Transportation 21, pp.107-133*.

Hensher, D.A. (2008) Hypothetical bias and stated choice studies, submitted to *Transportation Research Part B*.

Hensher, D.A., Greene, W.H. and Rose, J.M. (2006) Deriving willingness to pay estimates of travel time savings from individual-based parameters, *Environment and Planning A*, 38, 2365-2376.

Hensher, D. & Bradley, M. (1993) Using stated response choice data to enrich revealed preference discrete choice models. *Marketing Letters,* 4**,** 2**,** 139-151.

Hess, S. and Rose, J.M. (2009a) Allowing for intra-respondent variations in coefficients estimated on stated preference data, *Transportation Research Part B*, 43(6), 708-719.

Hess, S. and Rose, J.M. (2009b) Should reference alternatives in pivot design SC surveys be treated differently?, *Environment and Planning A*, 42(3), 297-317.

Hess, S., Smith, C., Falzarano, S. & Stubits, J. (2008) Measuring the effects of different experimental designs and survey administration methods using an Atlanta Managed Lanes Stated Preference survey, *Transportation Research Record*, 2049, 144-152.

Hess, S., Rose, J.M. & Polak, J.W. (2009), Non-trading, lexicographic and inconsistent behaviour in stated choice data, Transportation Research Part D, accepted for publication, January 2009.

Huber, J. and Zwerina, K. (1996). The importance of utility balance in efficient choice designs, *Journal of Marketing Research* 33: 307-317.

Kessels, R., Goos, P. and Vandebroek, M. (2006). A comparison of criteria to design efficient choice experiments, *Journal of Marketing Research* 43: 409-419.

Kuhfeld, W. F. and Tobias, R. D. (2005). Large factorial designs for product engineering and marketing research applications, *Technometrics* 47: 132-141.

Kuhfeld, W.F. (2005), “Marketing Research Methods in SAS”, *SAS Technical Report* (http://support.sas.com/techsup/technote/ts722.pdf).

Kuhfeld, W.F., Tobias, R.D., Garratt, M. (1994), “Efficient Experimental Design with Marketing Research Applications”, *Journal of Marketing Research*, vol. 31, pp. 545-557.

Kuzmanovic, M., Vukmirovia, D., (2005): The algorithms for constructing efficient experimental designs in conjoint analysis. The 7th Balkan conference on operation research, BACOR 05, Constanta

Lazari, Andreas G. and Don A. Anderson (1994), "Designs of Discrete Choice Set Experiments for Estimating Both Attribute and Availability Cross Effects," *Journal of Marketing Research,* 31 (August), 375-83.

Louviere, J.J and Timmermans H. (1990). Stated preference and choice models applied to recreation research: a review, Leisure Sciences, (12), pp 9-32

McFadden, Daniel (1974), "Conditional LogitAnalysis of Qualitative Choice Behavior," in *Frontiers in Econometrics,* P. Zarembka, ed. New York: Academic Press, 105-42.

Mitchell, Toby J. (1974), "An Algorithm for the Construction of D-optimal Experimental Designs," *Technometrics,* 16 (May), 203-10.

Ngene 1.1 User manual and reference guide (2011): The cutting edge in experimental design (http://www.choice-metrics.com)

Rardin, R.L., and Uzsoy, R. (2001): Experimental evaluation of heuristic optimization algorithms: A tutorial. Journal of Heuristics, 7:261-304

Rose, J.M. and Bliemer, M.C. (2006) Designing Efficient Data for Stated Choice Experiments, *presented at 11th International Conference on Travel Behaviour Research*, Kyoto, August 16-20, 2006, Japan.

Rose, J.M., Bliemer, M.C (2007) Designing stated choice experiments: State-of-the-Art. Seminar paper (*www.ivt.ethz.ch/news/archive/20071205_seminar_rose.pdf*)

Rose, J.M., Bliemer, M.C., Hensher and Collins, A. T. (2008) Designing efficient stated choice experiments in the presence of reference alternatives, *Transportation Research Part B*, 42(4), 395-406.

Rose, J.M. and Hess, S. (2009) Dual Response Choices In Reference Alternative Related Stated Choice Experiments, Transportation Research Board Annual Meeting, Washington DC January.

Sandor, Z. and Train, K. (2004). Quasi-random simulation of discrete choice models, *Transportation Research B* 38: 313-327.

Sandor, Z. and Wedel, M. (2001). Designing conjoint choice experiments using managers' prior beliefs, *Journal of Marketing Research* 38: 430-444.

Sandor, Z. and Wedel, M. (2002). Profile construction in experimental choice designs for mixed logit models, *Marketing Science* 21: 455-475.

Sandor, Z. and Wedel, M. (2005). Heterogeneous conjoint choice designs, *Journal of Marketing Research* 42: 210-218.

Toner, Clark, Grant-Muller, and Fowkes (1998): Anything you can do, we can do better. A provocative introduction to a new approach to stated preference design, CTR 8 at Anterp.

Toner, Wardman and Whelan (1999): Testing recent advances in stated preference design, European Transport Conference, Steam F, Cambridge

Tang, B. (1993). Orthogonal array-based Latin hypercubes, *Journal of the American Statistical Association* 88: 1392-1397.

Train, K. E. (2009). *Discrete Choice Methods with Simulation*, Cambridge U.K.: Cambridge University Press.

Timmermans H J P, 1984c, "Discrete choice models versus decomposition multiattribute preference models: a comparative analysis of model performance in the context of spatial shopping-behaviour", in London Papers in Regional Science 14: Discrete Choice Models in Regional Science Ed. DE Pitfield (Pion, London) pp88-102