KATHOLIEKE
UNIVERSITEIT
LEUVEN

DEPARTEMENT TOEGEPASTE ECONOMISCHE WETENSCHAPPEN

RESEARCH REPORT 0126

AN ADJUSTMENT ALGORITHM FOR THE CONSTRUCTION OF OPTIMAL RUN ORDERS

by
L. TACK
M. VANDEBROEK

D/2001/2376/26

An adjustment algorithm for the construction of optimal run orders

L. Tack
M. Vandebroek
Katholieke Universiteit Leuven, Belgium

Using a systematic run order can be the proper way to conduct an experiment when a temporal trend is present. The construction of run orders that are optimally balanced for time trend effects is based on the maximization of the information on the important parameters whereas the parameters of the postulated time trend are treated as nuisance parameters. In this paper, an adjustment algorithm is presented to improve the efficiency of the run orders obtained from a search over a predefined set of candidate points. This is done by repeatedly moving the design points or the time points of the candidate list a small amount along their axes as long as an improvement in the efficiency is obtained. It is illustrated that the adjustment algorithm involves substantial increases in the efficiency of the run orders. The use of the adjustment algorithm in addition to a search over a coarse grid of candidate points is especially recommended in situations where the computation time has to be kept within reasonable limits.

Keywords: $\mathcal{D}_t$-optimality; adjustment algorithm; run order; time trend

1 Introduction

Performing a series of measurements in a time sequence possibly creates time order dependence in the results. For example, when a batch of material is created at the beginning of an experiment and the treatments are to be applied to the experimental units formed from the material over time, there could be an unknown effect due to aging of the material. Unwanted time trend effects can also be caused by poisoning of a catalyst, steady build-up of deposits in a test engine, equipment wear-out, analyst fatigue, warm-up in laboratories, etc. For instance, Freeny and Lai (1997) mention an experiment taken from the electronics industry in which a photolithographic polisher shows the tendency to drift lower through time. Another example comes from Joiner and Campbell (1976) who describe an experiment in which the measurements drift with time due to the build-up of carbon in a spectrophotometer.

The next section shortly reviews the literature on the existence and the construction of run orders that are optimally balanced for time trend effects. Section 3 considers the problem in light of optimal design theory. Section 4 presents the adjustment algorithm and in section 5 some examples are given to illustrate the use of the algorithm.

2 Trend-free run orders

When the experimenter has knowledge about the nature of the time trend it is recommended to construct a run order in which the estimates of the treatment differences or the factorial effects are little disturbed by the presence of the time trend. This time dependence is usually approximated by a polynomial function of order q. Let $\mathbf{g}(t)$ be the $q \times 1$ vector representing the polynomial expansion for the time trend, expressed as a function of time $t \in [-1, 1]$. With $\boldsymbol{\beta}$ the $q \times 1$ vector of parameters of the polynomial time trend, $\mathbf{f}(\mathbf{x})$ the $p \times 1$ vector representing the polynomial expansion of $\mathbf{x} = [x_1, \ldots, x_f]'$ for the response and $\boldsymbol{\alpha}$ the $p \times 1$ vector of important parameters, let the statistical model be

$$y = \mathbf{f}'(\mathbf{x})\boldsymbol{\alpha} + \mathbf{g}'(t)\boldsymbol{\beta} + \varepsilon. \quad (1)$$

The error terms are assumed to be independent and identically distributed with mean zero and constant variance σ_ε^2. Contrary to simply observing the system twice for successive replicate points, it is understood that the transition from one run to the next one involves some intervention. As a result, two observations on the same design point are clearly subject to different errors from two runs and the assumption of uncorrelated error terms is justified. The fact that there is no interaction between the controllable variables x_i, with $i \in \{1, \ldots, f\}$, and time t usually holds in practice. For n observations, it is convenient to rewrite (1) as

$$\mathbf{y} = \mathbf{F}\boldsymbol{\alpha} + \mathbf{G}\boldsymbol{\beta} + \boldsymbol{\varepsilon}, \quad (2)$$

where $\mathbf{F}$ and $\mathbf{G}$ represent the $n \times p$ and the $n \times q$ design matrices respectively. The least-squares estimators $\hat{\boldsymbol{\alpha}}$ are given by

$$\hat{\boldsymbol{\alpha}} = [\mathbf{F}'\{\mathbf{I}_n - \mathbf{G}(\mathbf{G}'\mathbf{G})^{-1}\mathbf{G}'\}\mathbf{F}]^{-1}\mathbf{F}'\{\mathbf{I}_n - \mathbf{G}(\mathbf{G}'\mathbf{G})^{-1}\mathbf{G}'\}\mathbf{y}, \tag{3}$$

where $\mathbf{I}_n$ is the n-dimensional unity matrix. The factorial effects are now said to be q-trend-free if the least-squares estimator $\hat{\boldsymbol{\alpha}}$ is the same as when the time trend of qth order is not present. It is easy to verify that the least-squares estimator (3) is equal to the estimator $\hat{\boldsymbol{\alpha}} = (\mathbf{F}'\mathbf{F})^{-1}\mathbf{F}'\mathbf{y}$ in the absence of trend effects if and only if $\mathbf{F}'\mathbf{G} = \mathbf{0}$. Consequently, a run order is called q-trend-free if the columns of $\mathbf{F}$ are orthogonal to the columns of $\mathbf{G}$.

Cox (1951) was the first to study the construction of run orders for the estimation of treatment effects in the presence of a polynomial time trend. He presents trend-free run orders for a number of simple design problems for categorical variables. The most important contributions to trend-resistant experimental design for quantitative variables come from Cheng (1985) and Daniel and Wilcoxon (1966). Cheng (1985) formulates the Generalized Foldover Scheme with which generator sets can be created to construct trend-free run orders of full or fractional factorial designs. Based on the trend-free properties of the standard run order of factorial designs, Daniel and Wilcoxon (1966) develop another method for the construction of trend-free run orders but every run order derived with their method can also be obtained by applying the Generalized Foldover Scheme. A more extensive survey of the literature on trend-resistant design of experiments is given in Tack and Vandebroek (2001).

3 $\mathcal{D}_t$-optimal run orders

Although the above references on the existence and the construction of trend-free run orders are of great use to the practitioner, they all have a number of important shortcomings. Firstly, none of the references deals with the numerous situations in which the required orthogonality cannot be attained. Secondly, there is no consideration of the connection between trend-resistance and the generalized variance or the precision of the parameter estimates. Thirdly, all approaches assume that the treatments or factor level combinations have already been chosen. Finally, the approaches rely on assumptions that simplify reality too much. For instance, it is assumed that the number of time points is equal to the number of observations and that the time points are equally spaced. Besides, only 2- and 3-level factorial designs are considered, interactions of higher order are assumed to be negligible and constrained experimental regions cannot be dealt with.

To fill these gaps, Atkinson and Donev (1996) present a generic exchange algorithm to compute run orders that are optimally balanced for time trends. Tack and Vandebroek (2001) extend the exchange algorithm to allow for cost considerations and they develop a methodology to construct run orders that maximize the experimental efficiency defined as the amount of information obtained per unit cost, rather than focusing too much on a purely statistical efficiency. Contrary to offering a catalogue of specific solutions to design problems with a special structure, they provide the experimenter with a broadly applicable method to tackle a wide range of practical design problems.

When primary interest is in the precision of the parameter estimates, the algorithm computes optimal run orders by maximizing the information on the important parameters $\boldsymbol{\alpha}$ whereas the parameters $\boldsymbol{\beta}$ of the time trend are treated as nuisance parameters. The resulting run order is referred to as the $\mathcal{D}_t$-optimal run order $\delta_{\mathcal{D}_t}$ and the optimality criterion equals

$$\mathcal{D}_t = \frac{\begin{vmatrix} \mathbf{F}'\mathbf{F} & \mathbf{F}'\mathbf{G} \\ \mathbf{G}'\mathbf{F} & \mathbf{G}'\mathbf{G} \end{vmatrix}}{|\mathbf{G}'\mathbf{G}|} = |\mathbf{F}'\mathbf{F} - \mathbf{F}'\mathbf{G}(\mathbf{G}'\mathbf{G})^{-1}\mathbf{G}'\mathbf{F}|. \tag{4}$$

In the absence of time trend effects, the $\mathcal{D}$-optimal design $\delta_{\mathcal{D}}$ maximizes the determinant of the information matrix, i.e. $\mathcal{D} = |\mathbf{F}'\mathbf{F}|$. The $\mathcal{D}$-optimal design and the $\mathcal{D}_t$-optimal run order are then compared to each other through the trend factor

$$\mathrm{TF}(\delta_{\mathcal{D}_t}) = \left\{ \frac{\mathcal{D}_t(\delta_{\mathcal{D}_t})}{\mathcal{D}(\delta_{\mathcal{D}})} \right\}^{1/p}. \tag{5}$$

The power $1/p$ assures that the trend factor is independent of the dimension of the model. This means for instance that a run order $\delta_{\mathcal{D}_t}$ with trend factor 0.5 has to be replicated twice in order to be equally informative as the $\mathcal{D}$-optimal design $\delta_{\mathcal{D}}$. It follows readily that a trend-free run order, i.e. a run order with $\mathbf{F}'\mathbf{G} = \mathbf{0}$, attains the maximum value $\mathrm{TF}(\delta_{\mathcal{D}_t}) = 1$. In situations where it is impossible to obtain completely trend-free run orders, $\mathrm{TF}(\delta_{\mathcal{D}_t})$ will be less than 1.

4 An adjustment algorithm

The exchange algorithm of Tack and Vandebroek (2001) computes $\mathcal{D}_t$-optimal run orders by the allocation of n observations selected from a candidate list of d distinct design points to n out of h available time points in such a way as to maximize optimality criterion (4). The number n and the list of time points are user-specified. The candidate set of design points is also user-specified or can be computed as follows: the number of equally spaced levels per factor equals two, three, four or more depending on the fact whether the polynomial expansion $\mathbf{f}(\mathbf{x})$ is of first, second, third or higher order respectively. For instance, for a second-order polynomial the three factor levels under study are -1, 0 and 1. These three factor levels are commonly encountered in response surface methodology. This is due to two main reasons. Firstly, Farell et al. (1967) demonstrate that the factor levels of the optimal continuous design for a second-order polynomial over a hypercubic experimental region are -1, 0 and 1. Secondly, many practitioners prefer to use only a small number of distinct factor levels in order to keep the experimental effort within reasonable limits.

However, optimal design theory has shown that exact optimal designs found by searching over a continuous experimental region often contain other factor levels than -1, 0 and 1. For instance, in the absence of trend effects, Box and Draper (1971) and Donev and Atkinson (1988) show that for some factors other levels than -1, 0 and 1 lead to more efficient exact designs. Box and Draper (1971) analytically compute exact $\mathcal{D}$-optimal designs for second-order response models whereas Donev and Atkinson (1988) present an adjustment algorithm to improve the efficiency of $\mathcal{D}$-optimal designs with or without fixed block effects. This is done by moving the design points along their factor axes as long as an increase in the $\mathcal{D}$-efficiency is achieved. For a second-order polynomial model with uncorrelated observations and the number of factors up to five, Donev and Atkinson (1988) and Atkinson and Donev (1992) show that the effect of their adjustment algorithm is largest when the number of observations is equal to or just larger than the number of parameters p. For instance, for a second-order polynomial in three factors and $n = p = 10$, they obtain improvements in the $\mathcal{D}$-efficiency of about 3%. For a second-order model with fixed block effects and one or two factors, they show that substantial gains can be realized when there are only a few observations per block.

Goos (2001) investigates the gains obtained by using an adjustment algorithm to improve the $\mathcal{D}$-efficiency of a number of different designs with fixed or random block effects. For a full second-order response model, improvements of up to 10% are obtained. On the whole, the efficiency gains of his adjustment algorithm are superior to those obtained by searching over a fine hypercubic grid with 11 equally spaced levels for each factor. The best efficiencies are achieved by using both a fine grid and the adjustment algorithm. Generally speaking, the combined approach gives the best results for small n, small block sizes and highly correlated observations.

The previous references have clearly illustrated the large efficiency gains that result from using an adjustment algorithm. For that reason, the next section will present an adjustment algorithm to improve the trend factors of the $\mathcal{D}_t$-optimal run orders computed with the exchange algorithm of Tack and Vandebroek (2001).

4.1 Description of the algorithm

It is important to point out that the adjustment algorithm to be outlined in this section is a generalization of the adjustment algorithms of Donev and Atkinson (1988) and Goos (2001) because the time points can be adjusted too. The adjustment algorithm computes the effect on the $\mathcal{D}_t$-optimality criterion of moving any design point a small amount—henceforth referred to as the step length—along its factor axes and of moving any time point a small step along the time axis. As a matter of fact, interest is only in design changes that lead to design points still lying within the experimental region and time points still belonging to [-1,1]. This means that at most $2nf$ modifications for the design points and $2n$ changes for the time points have to be investigated. The design modification that leads to the largest increase in the criterion value is carried out and the process is repeated until no further progress can be made. The step length is then halved and the improvement process starts again. The algorithm attains its final iteration when the step lengths for the design points and the time points become smaller than their user-specified minimum value.

The input to the algorithm consists of the number of factors f, the order and the number of parameters p of the response model, the polynomial expansion for the response model $\mathbf{f}(\mathbf{x})$, the order and the number of parameters q of the time trend, the polynomial expansion for the time trend $\mathbf{g}(t)$ and the number of observations n. The initial step length for the design points and the initial step length for the time points, henceforth denoted as S_1 and S_2 respectively, have to be supplied too. The minimum step length for the design points and the minimum step length for the time points, referred to as $s_{1,\min}$ and $s_{2,\min}$ respectively, are also user-specified. Another input parameter specifies the design changes to be evaluated: only design points considered to be moved, only time points to be moved or both design points and time points allowed to be changed. The first option is chosen when the time points are fixed by the design problem at hand whereas the second option refers to the situation in which the different factor level combinations have already been chosen. The experimenter has the additional possibility to impose a minimum distance—in the sequel of this paper denoted as MD—to be maintained between two successive time points. Such a situation occurs when a fixed measurement time has to be taken into account. Finally, the input also contains the $\mathcal{D}_t$-optimal run order to be used as the starting run order in the adjustment algorithm. The appendix gives a formal outline of the adjustment algorithm.

4.2 Update formulae

A considerable reduction in the computation time is achieved when powerful update formulae are used to evaluate the effect of moving a time point or a co-ordinate of a design point along its axis. This is thanks to the fact that the matrices in the numerator and the denominator of optimality criterion (4) can be written as a sum of outer products

$$\mathbf{H}'\mathbf{H} = \sum_{\forall(\mathbf{x}_i,t_k)} \mathbf{h}(\mathbf{x}_i,t_k)\mathbf{h}'(\mathbf{x}_i,t_k), \tag{6}$$

where $\mathbf{h}(\mathbf{x}_i,t_k) = [\mathbf{f}'(\mathbf{x}_i)\ \mathbf{g}'(t_k)]'$ or $\mathbf{h}(\mathbf{x}_i,t_k) = \mathbf{g}(t_k)$ for the numerator or the denominator of (4) respectively. Let the current step length for the design points and the time points be written as s_1 and s_2 respectively. After moving the rth factor of design point $\mathbf{x}_j$ from level x_{jr} to level $x_{jr} \pm s_1$ and time point t_l to $t_l \pm s_2$, (6) becomes

$$\mathbf{H}'\mathbf{H} - \mathbf{h}(\mathbf{x}_j,t_l)\mathbf{h}'(\mathbf{x}_j,t_l) + \mathbf{h}(\mathbf{x}_j \pm s_1\mathbf{e}_r, t_l \pm s_2)\mathbf{h}'(\mathbf{x}_j \pm s_1\mathbf{e}_r, t_l \pm s_2), \tag{7}$$

where the f-dimensional column vector $\mathbf{e}_r$ contains element 1 at position r and zeros elsewhere. According to theorem 18.1.1 of Harville (1997), the determinant of (7) can be written as

$$
\begin{aligned}
&|\mathbf{H}'\mathbf{H}|[\{1-\mathbf{h}'(\mathbf{x}_j,t_l)(\mathbf{H}'\mathbf{H})^{-1}\mathbf{h}(\mathbf{x}_j,t_l)\}\\
&\quad\times\{1+\mathbf{h}'(\mathbf{x}_j\pm s_1\mathbf{e}_r,t_l\pm s_2)(\mathbf{H}'\mathbf{H})^{-1}\mathbf{h}(\mathbf{x}_j\pm s_1\mathbf{e}_r,t_l\pm s_2)\}\\
&\quad+\{\mathbf{h}'(\mathbf{x}_j\pm s_1\mathbf{e}_r,t_l\pm s_2)(\mathbf{H}'\mathbf{H})^{-1}\mathbf{h}(\mathbf{x}_j,t_l)\}^2].
\end{aligned}
\tag{8}
$$

Based on (8), the effect on the criterion value (4) of adjusting a design point or a time point can readily be calculated without the need for computationally intensive determinant operations.

5 Illustrative examples

This section illustrates the utility of the adjustment algorithm described in section 4. The first example concerns quadratic regression in one explanatory variable. In the second example, adjusted run sequences for the 2^4 full factorial design are constructed. The full second-order polynomial in three factors is investigated in the third example. The final example traces whether the adjustment algorithm serves as a valid alternative to the exchange algorithm of Tack and Vandebroek (2001) when the total computation time has to be kept low.

5.1 Quadratic regression in one variable

Consider the problem of designing an experiment for the estimation of a quadratic model $\mathbf{f}(\mathbf{x}) = [1, x, x^2]'$ and coded factor levels -1, 0 and 1. Use is made of the exchange algorithm of Tack and Vandebroek (2001) to compute $\mathcal{D}_t$-optimal run orders for postulated time trends of first, second, third and fourth order, denoted as $\mathbf{g}_1(t)$, $\mathbf{g}_2(t)$, $\mathbf{g}_3(t)$ and $\mathbf{g}_4(t)$ respectively. The design sizes equal $n = 7$, $n = 8$, $n = 9$ or $n = 10$ and there are as much equally spaced time points between -1 and 1. The trend factors of the computed $\mathcal{D}_t$-optimal run orders $\delta_{\mathcal{D}_t}$ are given in table 1 in the columns with label TF($\delta_{\mathcal{D}_t}$). For instance, for $n = 7$ and $n = 9$ the $\mathcal{D}_t$-optimal run orders are completely trend-free for a linear time trend $\mathbf{g}_1(t)$. However, the $\mathcal{D}_t$-optimal run orders are poorly balanced for higher-order time trends.

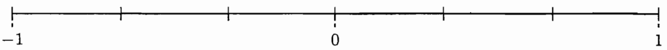
It will now be investigated whether an adjustment of the time points leads to an increase in the trend factors of the $\mathcal{D}_t$-optimal run orders or not. Therefore, the adjustment algorithm is used with the initial step length for the time points $S_2 = 2$, the minimum step length $s_{2,\min} = 10^{-5}$ and the minimum distance MD $= 10^{-5}$. The trend factors of the adjusted run orders δ_{adj} are also given in table 1. It can easily be seen that considerable increases in the trend factors are achieved. For instance, adjusting the time points of the $\mathcal{D}_t$-optimal run order for $n = 7$ and time trend $\mathbf{g}_2(t)$ augments the trend factor from 0.712 to 0.752. This is an improvement of about 5.6%. The increase in the trend factor is largest for the run orders computed with time trend $\mathbf{g}_4(t)$. In that case improvements of up to 30% are observed.

Table 1: Trend factors of the $\mathcal{D}_t$-optimal run orders and the adjusted run orders for quadratic regression in one variable and different numbers of observations

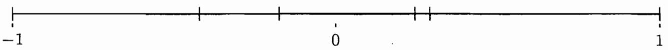
	$n=7$		$n=8$		$n=9$		$n=10$	
	TF($\delta_{\mathcal{D}_t}$)	TF(δ_{adj})	TF($\delta_{\mathcal{D}_t}$)	TF(δ_{adj})	TF($\delta_{\mathcal{D}_t}$)	TF(δ_{adj})	TF($\delta_{\mathcal{D}_t}$)	TF(δ_{adj})
$\mathbf{g}_1(t)$	1.000	1.000	0.999	1.000	1.000	1.000	0.999	1.000
$\mathbf{g}_2(t)$	0.712	0.752	0.743	0.817	0.753	0.818	0.754	0.846
$\mathbf{g}_3(t)$	0.677	0.689	0.706	0.763	0.705	0.763	0.731	0.793
$\mathbf{g}_4(t)$	0.451	0.591	0.545	0.688	0.559	0.689	0.579	0.725

As an illustration, the time points of the adjusted run orders for $n = 7$, $n = 8$, $n = 9$ and $n = 10$ are given in figure 1 to figure 4 respectively. The adjusted time points are indicated by means of small vertical dashes. For a linear time trend $\mathbf{g}_1(t)$ and any number of observations n, the time points are approximately uniformly spread over the entire time range [-1,1]. This observation closely resembles the usual assumption of equally spaced time points between -1 and 1. However, the figures reveal that this assumption is no longer optimal for higher-order time trends. For instance, it can clearly be seen that for time trend $\mathbf{g}_4(t)$ the time points are more concentrated around $t = 0$ and the boundaries $t = -1$ and $t = 1$.

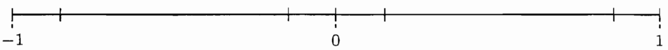
(a) $\mathbf{g}_1(t)$



(b) $\mathbf{g}_2(t)$



(c) $\mathbf{g}_3(t)$



(d) $\mathbf{g}_4(t)$

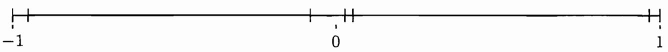


Figure 1: Time points of the adjusted run orders for quadratic regression in one variable and $n = 7$

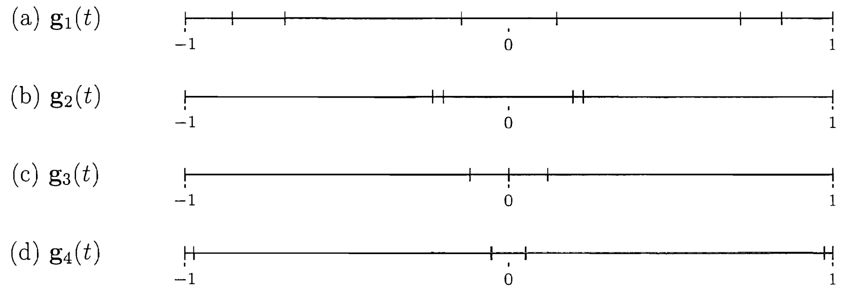


Figure 2: Time points of the adjusted run orders for quadratic regression in one variable and $n = 8$

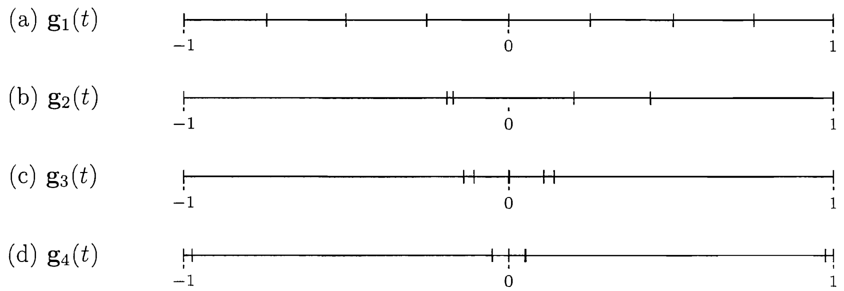


Figure 3: Time points of the adjusted run orders for quadratic regression in one variable and $n = 9$

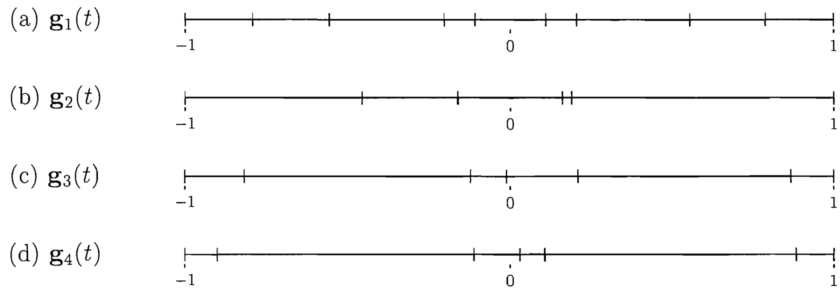


Figure 4: Time points of the adjusted run orders for quadratic regression in one variable and $n = 10$

It is interesting to note that in almost every case an adjustment of the design points -1, 0 and 1 of the $\mathcal{D}_t$-optimal run orders also has a positive effect on the criterion value. For instance, whereas the $\mathcal{D}$-optimal design for a quadratic regression in one variable and $n = 9$ observations has three observations at each of the levels -1, 0 and 1, the corresponding adjusted run order for time trend $\mathbf{g}_4(t)$ possesses the levels -1, -0.297, -0.016, 0.063 and 1. As compared to the $\mathcal{D}_t$-optimal run order for the candidate set -1, 0 and 1, the improvement in the trend factor amounts to 0.7%. Although the improvements are very small, the important conclusion is that trend-robust run orders do not necessarily have the same support as their $\mathcal{D}$-optimal counterparts.

5.2 The 2^4 factorial design

The aim of this example is to verify whether the benefit of the adjustment algorithm can be generalized to any initial step length S_2 and minimum distance MD. This example starts with the computation of $\mathcal{D}_t$-optimal run orders for the 2^4 full factorial design with polynomial expansion

$$\mathbf{f}(\mathbf{x}) = [1, x_1, x_2, x_3, x_4, x_1x_2, x_1x_3, x_1x_4, x_2x_3, x_2x_4, x_3x_4]',$$

$2^4 = 16$ observations, as much equally spaced time points between -1 and 1 and postulated time trends $\mathbf{g}_1(t)$, $\mathbf{g}_2(t)$, $\mathbf{g}_3(t)$ and $\mathbf{g}_4(t)$. The trend factors of the $\mathcal{D}_t$-optimal run orders are 1, 0.900, 0.849 and 0.758 respectively. It follows that for time trends of order two or higher complete trend-resistance cannot be attained.

The adjustment algorithm is again used to trace whether improvements in the trend factors of the $\mathcal{D}_t$-optimal run orders for time trends $\mathbf{g}_2(t)$, $\mathbf{g}_3(t)$ and $\mathbf{g}_4(t)$ can be obtained. When only the design points of the $\mathcal{D}_t$-optimal run orders are subject to adjustment, no improvements in the trend factors could be achieved. This result does not come as a surprise since the 2^4 full factorial design is a $\mathcal{D}$-optimal design. An adjustment of the time points is also investigated and the results are shown in table 2.

Table 2: Trend factors of the $\mathcal{D}_t$-optimal run orders and the adjusted run orders for the 2^4 full factorial design and initial step length $S_2 = 0.10$

		TF(δ_{adj}); MD =					
	TF($\delta_{\mathcal{D}_t}$)	0.00	0.02	0.04	0.06	0.08	0.10
$\mathbf{g}_2(t)$	0.900	0.903	0.903	0.903	0.903	0.903	0.902
$\mathbf{g}_3(t)$	0.849	0.871	0.868	0.865	0.863	0.861	0.858
$\mathbf{g}_4(t)$	0.758	0.808	0.803	0.794	0.790	0.786	0.778

Each row refers to the adjusted run orders for a particular time trend and displays the trend factors for several minimum distances MD and initial step lengths $S_2 = 0.10$. It is important to note that the results are independent of the initial step length S_2. The minimum step length for the time points was set equal to $s_{2,\text{min}} = 10^{-5}$. The table shows that for all cases investigated an adjustment of the time points leads to an improvement in the trend factor. Consequently, it again follows that the usual assumption of equally spaced time points is not always the best one. For instance, for a second-order time trend $\mathbf{g}_2(t)$ and minimum distance MD equal to 0.00, the upper left cell shows that the trend factor of the adjusted run order is 0.903. As compared to the trend factor 0.900 of the corresponding $\mathcal{D}_t$-optimal run order, this is only a small improvement. The increases in the trend factor are more pronounced as the order of the postulated time trend grows larger. For instance, for time trend $\mathbf{g}_4(t)$ and minimum distance MD $= 0.00$, improvements of more than 6.5% are achieved. As a matter of fact, the smaller the minimum distance to be maintained between two successive time points, the larger the improvement in the trend factor.

As an illustration, for the postulated time trends $\mathbf{g}_2(t)$, $\mathbf{g}_3(t)$ and $\mathbf{g}_4(t)$, the time points of the adjusted run orders are displayed in figure 5, figure 6 and figure 7 respectively. The adjusted time points are again indicated by means of small vertical dashes. As a matter of fact, the smaller the minimum distance to be maintained between successive time points, the less the adjusted time points are equally spaced.

5.3 The 3^3 factorial design

In this section the effect of adjusting the design points will be investigated. Attention is restricted to experiments in which three factors are presumed to influence the response of interest, the polynomial expansion

$$\mathbf{f}(\mathbf{x}) = [1, x_1, x_2, x_3, x_1x_2, x_1x_3, x_2x_3, x_1^2, x_2^2, x_3^2]',$$

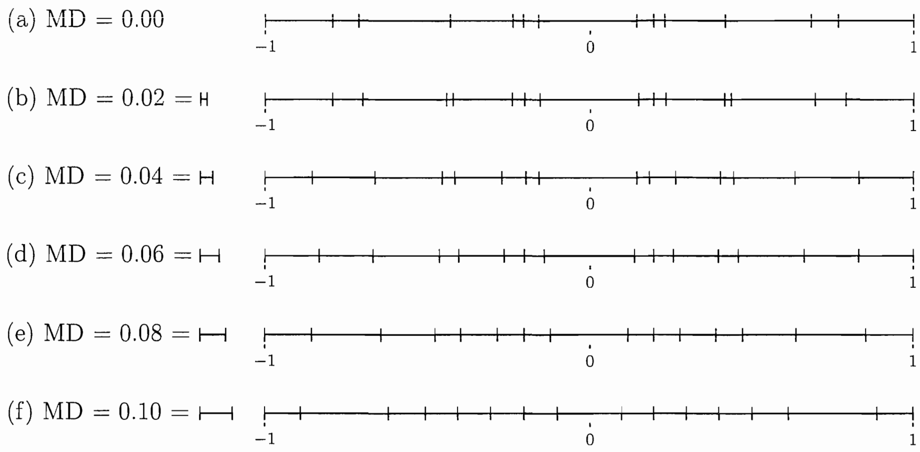


Figure 5: Time points of the adjusted run orders for the 2^4 full factorial design, time trend $\mathbf{g}_2(t)$ and initial step length 0.10

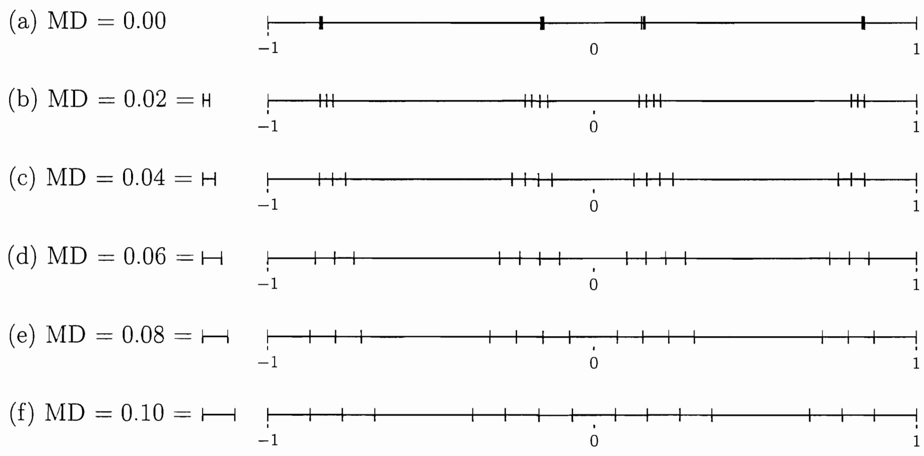


Figure 6: Time points of the adjusted run orders for the 2^4 full factorial design, time trend $\mathbf{g}_3(t)$ and initial step length 0.10

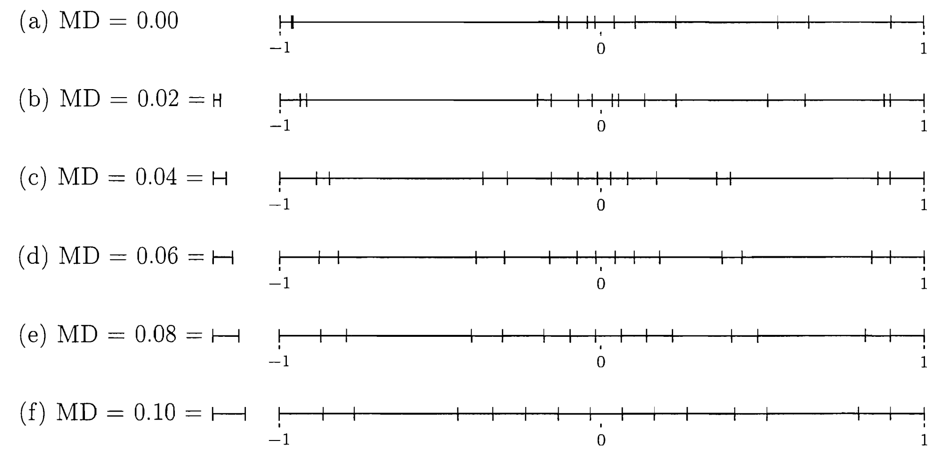


Figure 7: Time points of the adjusted run orders for the 2^4 full factorial design, time trend $\mathbf{g}_4(t)$ and initial step length 0.10

$3^3 = 27$ observations and $h = 27$ equally spaced time points between -1 and 1. The postulated time trends are again written as $\mathbf{g}_1(t)$, $\mathbf{g}_2(t)$, $\mathbf{g}_3(t)$ and $\mathbf{g}_4(t)$. The trend factors of the $\mathcal{D}_t$-optimal run orders for the full 3^3 factorial design equal 0.9413, 0.8677, 0.8663 and 0.8230 for the respective time trends.

In the sequel of this example, three different approaches will be investigated to improve the trend-resistance of the $\mathcal{D}_t$-optimal run orders of the 3^3 full factorial design. Firstly, the adjustment algorithm is applied to the $\mathcal{D}_t$-optimal run orders computed with $d = 27$ and $h = 27$. The initial step lengths for the design points and the time points are set equal to $S_1 = 0.5$ and $S_2 = 0.05$ respectively. Since the design points of the 3^3 factorial design do not constitute a $\mathcal{D}$-optimal design, improvements in the trend factor can be expected if the design points are considered to be adjusted. The results are given in table 3. Each panel refers to the run orders for a particular time trend. The last three lines of each panel are related to the adjusted run orders. The table shows that for a linear time trend $\mathbf{g}_1(t)$ the adjustment algorithm does not lead to an improvement in the trend factor of the $\mathcal{D}_t$-optimal run order with $d = 27$ and $h = 27$. For higher-order time trends, the adjustment algorithm yields an improvement. For instance, adjusting the time points of the $\mathcal{D}_t$-optimal run order for time trend $\mathbf{g}_2(t)$, $d = 27$ and $h = 27$ gives a run order with trend factor 0.9004. The best results are obtained when both the time points and the design points are considered to be adjusted. In that case, improvements of up to 7% are achieved. This once more demonstrates that the $\mathcal{D}_t$-optimal run orders are in fact only optimal for the specified set of design points and time points.

Secondly, the use of a finer grid for the design points or the time points in the exchange algorithm is also studied. Three fine grids are investigated: one with $d = 27$ design points and $h = 41$ equally spaced time points between -1 and 1, one with $d = 125$ design points on a regular $5 \times 5 \times 5$ grid and $h = 27$ time points and one with $d = 125$ design points and $h = 41$ equally spaced time points. The trend factors of the corresponding $\mathcal{D}_t$-optimal run orders are given by the last three numbers in the fourth row of each panel. The largest improvement occurs for time trend $\mathbf{g}_4(t)$, $d = 125$ and $h = 41$ and is slightly more than 5%.

Finally, the best results are obtained when a search over a fine grid is combined with the adjustment algorithm. This is illustrated in the last three rows and columns of each panel. The most striking improvement is when both the time points and the design points are considered for adjustment. As compared to the trend factors of the $\mathcal{D}_t$-optimal run orders computed over the coarse grid with $d = 27$ and $h = 27$, improvements of up to 13% are observed.

5.4 Computational aspects of the adjustment algorithm

In this example focus is again on design problems for the second-order response function in three factors

$$\mathbf{f}(\mathbf{x}) = [1, x_1, x_2, x_3, x_1x_2, x_1x_3, x_2x_3, x_1^2, x_2^2, x_3^2]',$$

time trends $\mathbf{g}_1(t)$, $\mathbf{g}_2(t)$, $\mathbf{g}_3(t)$ and $\mathbf{g}_4(t)$, $n = 27$ observations and $h = 27$ equally spaced time points between -1 and 1. The design points are taken from the regular $3 \times 3 \times 3$ grid or from the regular $5 \times 5 \times 5$ grid. The number of different design points then equals $d = 3^3 = 27$ or $d = 5^3 = 125$ respectively. The exchange algorithm of Tack and Vandebroek (2001) is used to compute $\mathcal{D}_t$-optimal run orders $\delta_{\mathcal{D}_t}$ for both grids and the results are given in table 4. The first panel of the table shows the trend factors of the computed run orders, whereas the second panel displays the average computation times per try. The best protection against time trend effects is obtained when a fine grid for the candidate points is used (i.e. $d = 125$ instead of $d = 27$). For instance, the $\mathcal{D}_t$-optimal run order for a second-order time trend and $d = 27$ has trend factor 0.9217, whereas the trend factor of the corresponding run order for $d = 125$ has trend factor 0.9637. However, this increase in the protection against temporal dependence goes at the expense of the computation time; the computation of the $\mathcal{D}_t$-optimal run order with $d = 27$ takes on the average 6.2 seconds, whereas the computation time of the $\mathcal{D}_t$-optimal run order for $d = 125$ is about 100 times larger. It follows that for a fine grid of the design points (i.e. for large values of d) the exchange algorithm becomes very slow. Consequently, when the total computation time is an important issue to be considered, there is a strong need for a worthy alternative to compute trend-robust run orders within a reasonable computation time. It is now interesting to investigate whether the adjustment algorithm can be used for that purpose or not. The $\mathcal{D}_t$-optimal run orders for $d = 27$ are therefore used as the starting run orders in the adjustment algorithm. Only the design points can be adjusted and their initial step and minimum step are specified large enough to assure that the

Table 3: Trend factors of the $\mathcal{D}_t$-optimal run orders and the adjusted run orders for the full second-order response model in three factors and $n = 27$ and for different numbers of design points and time points

$g_1(t)$				
d	27	27	125	125
h	27	41	27	41
$\mathcal{D}_t$-optimal run order	0.9413	0.9413	0.9519	0.9519
adjustment of time points	0.9413	0.9413	0.9519	0.9519
adjustment of design points	0.9413	0.9413	0.9964	0.9956
adjustment of time points and design points	0.9413	0.9413	0.9969	0.9967
$g_2(t)$				
d	27	27	125	125
h	27	41	27	41
$\mathcal{D}_t$-optimal run order	0.8677	0.8992	0.8773	0.9097
adjustment of time points	0.9004	0.9109	0.9121	0.9213
adjustment of design points	0.8677	0.8993	0.9139	0.9461
adjustment of time points and design points	0.9275	0.9370	0.9600	0.9580
$g_3(t)$				
d	27	27	125	125
h	27	41	27	41
$\mathcal{D}_t$-optimal run order	0.8663	0.8895	0.8765	0.9005
adjustment of time points	0.8922	0.9005	0.9116	0.9139
adjustment of design points	0.8667	0.8899	0.9053	0.9402
adjustment of time points and design points	0.9172	0.9012	0.9390	0.9562
$g_4(t)$				
d	27	27	125	125
h	27	41	27	41
$\mathcal{D}_t$-optimal run order	0.8230	0.8552	0.8317	0.8652
adjustment of time points	0.8756	0.8797	0.8855	0.8924
adjustment of design points	0.8238	0.8561	0.8575	0.8919
adjustment of time points and design points	0.8776	0.8810	0.9159	0.9283

support points of the adjusted run orders form part of the regular $5 \times 5 \times 5$ grid. This assures a valid comparison between the adjusted run orders and the $\mathcal{D}_t$-optimal run orders for $d = 125$. The trend factors of the adjusted run orders δ_{adj} are shown in the last column of the first panel. For instance, the adjusted run order for $\mathbf{g}_2(t)$ has trend factor 0.9536, which is very close to trend factor 0.9637 of the $\mathcal{D}_t$-optimal run order computed with $d = 125$. Since the computation time for the adjustment algorithm is negligible to that of the exchange algorithm for large numbers of tries, the computation time of the adjusted run order is only a negligible amount larger than 6.2 seconds. This is much less than the 601.2 seconds needed for the corresponding $\mathcal{D}_t$-optimal run order. It follows that at least for $h = 27$ and a second-order time trend, the adjustment algorithm is very suitable to compute good run orders when the computation time is an important issue. A similar conclusion can be drawn for the other time trends.

Table 4: Trend factors and computation times of the $\mathcal{D}_t$-optimal run orders and the adjusted run orders for the full second-order response model in three factors, $n = 27$ and $h = 27$

	trend factor			computation time (sec)		
	$\delta_{\mathcal{D}_t}$ $(d = 27)$	$\delta_{\mathcal{D}_t}$ $(d = 125)$	δ_{adj}	$\delta_{\mathcal{D}_t}$ $(d = 27)$	$\delta_{\mathcal{D}_t}$ $(d = 125)$	δ_{adj}
$\mathbf{g}_1(t)$	1.0000	1.0000	1.0000	3.1	274.8	3.1
$\mathbf{g}_2(t)$	0.9217	0.9637	0.9536	6.2	601.2	6.2
$\mathbf{g}_3(t)$	0.9202	0.9501	0.9410	7.6	567.0	7.6
$\mathbf{g}_4(t)$	0.8690	0.9189	0.9045	7.7	544.7	7.7

The next example compares the adjustment of the $\mathcal{D}_t$-optimal run orders computed with $d = 27$ and $h = 27$ with the $\mathcal{D}_t$-optimal run orders for $d = 27$ and $h = 53$. The former ones are used as input to the adjustment algorithm and to make the comparison valid, only the time points are allowed to be adjusted. Besides, the parameters of the adjustment algorithm are specified such that the time points of the adjusted run orders form part of the set of time points of the $\mathcal{D}_t$-optimal run orders for $h = 53$. The results are given in table 5. For instance, the trend factor of the adjusted run order for a quadratic time trend is equal to 0.9536, which is again very close to trend factor 0.9638 of the corresponding $\mathcal{D}_t$-optimal run order with $h = 53$. The computation time of the adjusted run order is much lower than that of the $\mathcal{D}_t$-optimal run order, namely 6.2 seconds is almost negligible as compared to 131.7 seconds. Similar conclusions hold for the other trend functions.

As a final illustration consider the comparison between run orders with possibly replicated observations and run orders without replicated observations. The number of design points and time points equals $d = 27$ and $h = 53$ respectively. The results are given in table 6. Taking into account the possibility for replicated observations naturally involves an improvement in the balance for time trend effects. For instance, the trend factor of the $\mathcal{D}_t$-optimal run order for $\mathbf{g}_1(t)$ without replicated observations is equal to 0.9413,

Table 5: Trend factors and computation times of the $\mathcal{D}_t$-optimal run orders and the adjusted run orders for the full second-order response model in three factors, $n = 27$ and $d = 27$

	trend factor			computation time (sec)		
	$\delta_{\mathcal{D}_t}$ ($h = 27$)	$\delta_{\mathcal{D}_t}$ ($h = 53$)	δ_{adj}	$\delta_{\mathcal{D}_t}$ ($h = 27$)	$\delta_{\mathcal{D}_t}$ ($h = 53$)	δ_{adj}
$g_1(t)$	1.0000	1.0000	1.0000	3.1	38.2	3.1
$g_2(t)$	0.9217	0.9638	0.9536	6.2	131.7	6.2
$g_3(t)$	0.9202	0.9501	0.9410	7.6	99.4	7.6
$g_4(t)$	0.8690	0.9193	0.9075	7.7	100.3	7.7

whereas allowing replicated observations leads to a completely linear-trend free run order. However, taking into account replicated observations goes at the cost of the computation time. Applying the adjustment algorithm—with the parameters set to allow replicated observations—to the $\mathcal{D}_t$-optimal run order without replicates augments the trend factor to 0.9926, which approximates complete trend-resistance. The computation time of the adjusted run order is again much lower than that of the $\mathcal{D}_t$-optimal run order with replicated observations. The conclusions can be generalized to the higher-order time trends.

Table 6: Trend factors and computation times of the $\mathcal{D}_t$-optimal run orders and the adjusted run orders for the full second-order response model in three factors, $n = 27$, $d = 27$ and $h = 53$

	trend factor			computation time (sec)		
	$\delta_{\mathcal{D}_t}$ (no repl.)	$\delta_{\mathcal{D}_t}$ (repl.)	δ_{adj}	$\delta_{\mathcal{D}_t}$ (no repl.)	$\delta_{\mathcal{D}_t}$ (repl.)	δ_{adj}
$g_1(t)$	0.9413	1.0000	0.9926	3.8	38.2	3.8
$g_2(t)$	0.9090	0.9638	0.9544	14.0	131.7	14.0
$g_3(t)$	0.8959	0.9501	0.9327	14.2	99.4	14.2
$g_4(t)$	0.8682	0.9193	0.8978	11.3	100.3	11.3

Summing up, this example has clearly illustrated that it is recommended to use the adjustment algorithm in addition to the exchange procedure when the computation time is an important issue. The resulting outperformance in computation time goes at the expense of the trend-resistance but the loss is only moderate.

6 Conclusion

This paper has proposed an adjustment algorithm to improve the trend factors of the $\mathcal{D}_t$-optimal run orders computed with the exchange algorithm. This is done by moving the design points and/or the time points of the predefined candidate set along their axes. Only design modifications that improve the $\mathcal{D}_t$-criterion value are accepted. The computational results have shown that an adjustment of the design points and/or the time points considerably increases the protection against time dependent results. Besides, it has been illustrated that the assumption of equally spaced time points is in general not a good one and that the adjustment algorithm is very useful to compute the optimal time points. Finally, instead of using a conventional exchange procedure to compute run orders over a fine grid, the use of the adjustment algorithm in addition to an exchange procedure over a coarse grid is very recommendable in situations where the aim is to obtain good run orders at an acceptable computation time.

Appendix. The design algorithm

In the outline of the algorithm, the starting run order is written as $R = \{(\mathbf{x}_i, t_j)\}$ and the criterion value will be denoted as $\mathcal{Q}$. The initial step lengths for the design points and the time points are denoted as S_1 and S_2 respectively. The minimum step lengths are written as $s_{1,\min}$ and $s_{2,\min}$ respectively. Finally, $\nu_1 = 1$ indicates that design points are considered to be adjusted, whereas the opposite holds for $\nu_1 \neq 1$. In a similar way, the value of parameter ν_2 indicates whether the time points are considered for adjustment or not. The values of ν_1 and ν_2 are of course user-specified. The practitioner has the additional possibility to impose a minimum distance to be maintained between any two successive time points. To preserve clarity, this option is left out from the outline. The output of the algorithm consists of the adjusted run order R_{opt} and its corresponding criterion value $\mathcal{Q}_{\text{opt}}$. After reading the input, the algorithm proceeds as follows:

1. Set $s_1 = S_1$ and $s_2 = S_2$.

2. Compute the criterion value $\mathcal{Q}$ of the starting run order R.

3. Set $\mathcal{Q}_{\text{opt}} = \mathcal{Q}$ and $R_{\text{opt}} = R$.

4. Adjust the run order:

   (a) Set $\triangle_1 = 1$.

   (b) If $\nu_1 = 1$, then find the best change in design points:
   $\forall \mathbf{x}_i \in R$, $\forall j \in \{1, \ldots, f\}$:

   - If $x_{ij} - s_1 \geq -1$, compute the effect $\triangle$ on $\mathcal{Q}_{\text{opt}}$ of changing x_{ij} to $x_{ij} - s_1$. If $\triangle > \triangle_1$ then set $\triangle_1 = \triangle$, $\omega_{11} = -1$, $a_1 = i$ and $b_1 = j$.
   - If $x_{ij} + s_1 \leq 1$, compute the effect $\triangle$ on $\mathcal{Q}_{\text{opt}}$ of changing x_{ij} to $x_{ij} + s_1$. If $\triangle > \triangle_1$ then set $\triangle_1 = \triangle$, $\omega_{11} = 1$, $a_1 = i$ and $b_1 = j$.

   (c) Set $\triangle_2 = 1$.

   (d) If $\nu_2 = 1$, then find the best change in time points:
   $\forall t_k \in R$:

   - If $t_k - s_2 \geq -1$, compute the effect $\triangle$ on $\mathcal{Q}_{\text{opt}}$ of changing t_k to $t_k - s_2$. If $\triangle > \triangle_2$ then set $\triangle_2 = \triangle$, $\omega_{22} = -1$ and $c_2 = k$.
   - If $t_k + s_2 \leq 1$, compute the effect $\triangle$ on $\mathcal{Q}_{\text{opt}}$ of changing t_k to $t_k + s_2$. If $\triangle > \triangle_2$ then set $\triangle_2 = \triangle$, $\omega_{22} = 1$ and $c_2 = k$.

   (e) Set $\triangle_3 = 1$.

   (f) If $\nu_1 = 1$ and $\nu_2 = 1$, then find the best change in design points and time points:
   $\forall (\mathbf{x}_i, t_k) \in R$, $\forall j \in \{1, \ldots, f\}$:

- If $x_{ij} - s_1 \geq -1$ and $t_k - s_2 \geq -1$, compute the effect $\triangle$ on $\mathcal{Q}_{\text{opt}}$ of changing x_{ij} to $x_{ij} - s_1$ and t_k to $t_k - s_2$.
  If $\triangle > \triangle_3$ then set $\triangle_3 = \triangle$, $\omega_{31} = -1$, $\omega_{32} = -1$, $a_3 = i$, $b_3 = j$ and $c_3 = k$.
- If $x_{ij} - s_1 \geq -1$ and $t_k + s_2 \leq 1$, compute the effect $\triangle$ on $\mathcal{Q}_{\text{opt}}$ of changing x_{ij} to $x_{ij} - s_1$ and t_k to $t_k + s_2$.
  If $\triangle > \triangle_3$ then set $\triangle_3 = \triangle$, $\omega_{31} = -1$, $\omega_{32} = 1$, $a_3 = i$, $b_3 = j$ and $c_3 = k$.
- If $x_{ij} + s_1 \leq 1$ and $t_k - s_2 \geq -1$, compute the effect $\triangle$ on $\mathcal{Q}_{\text{opt}}$ of changing x_{ij} to $x_{ij} + s_1$ and t_k to $t_k - s_2$.
  If $\triangle > \triangle_3$ then set $\triangle_3 = \triangle$, $\omega_{31} = 1$, $\omega_{32} = -1$, $a_3 = i$, $b_3 = j$ and $c_3 = k$.
- If $x_{ij} + s_1 \leq 1$ and $t_k + s_2 \leq 1$, compute the effect $\triangle$ on $\mathcal{Q}_{\text{opt}}$ of changing x_{ij} to $x_{ij} + s_1$ and t_k to $t_k + s_2$.
  If $\triangle > \triangle_3$ then set $\triangle_3 = \triangle$, $\omega_{31} = 1$, $\omega_{32} = 1$, $a_3 = i$, $b_3 = j$ and $c_3 = k$.

5. Compute $\triangle = \max(\triangle_1, \triangle_2, \triangle_3)$.
6. If $\triangle > 1$ then
   (a) If $\triangle = \triangle_1$, then replace $x_{a_1 b_1}$ with $x_{a_1 b_1} + \omega_{11} s_1$ and go to (d).
   (b) If $\triangle = \triangle_2$, then replace t_{c_2} with $t_{c_2} + \omega_{22} s_2$ and go to (d).
   (c) If $\triangle = \triangle_3$, then replace $x_{a_3 b_3}$ with $x_{a_3 b_3} + \omega_{31} s_1$ and t_{c_3} with $t_{c_3} + \omega_{32} s_2$.
   (d) Update $\mathcal{Q}_{\text{opt}}$.
   (e) Go to 4.
7. Reduce the step lengths:
   (a) If $\nu_1 = 1$, $\nu_2 \neq 1$ and $s_1/2 \geq s_{1,\min}$, then
      i. Set $s_1 = s_1/2$.
      ii. Go to 4.
   (b) If $\nu_1 \neq 1$, $\nu_2 = 1$ and $s_2/2 \geq s_{2,\min}$, then
      i. Set $s_2 = s_2/2$.
      ii. Go to 4.
   (c) If $\nu_1 = 1$, $\nu_2 = 1$ and ($s_1/2 \geq s_{1,\min}$ or $s_2/2 \geq s_{2,\min}$), then
      i. If $s_1/2 \geq s_{1,\min}$ then set $s_1 = s_1/2$.
      ii. If $s_2/2 \geq s_{2,\min}$ then set $s_2 = s_2/2$.
      iii. Go to 4.
8. Write $\mathcal{Q}_{\text{opt}}$ and R_{opt}.

The algorithm is implemented in Fortran 77 and makes use of the library Netlib of Bell Labs. It uses predefined routines to factor a symmetric matrix, to compute the determinant of a factored symmetric matrix, to invert a symmetric matrix and to generate random numbers.

References

Atkinson, A. and Donev, A. (1992). *Optimum Experimental Designs*, Oxford: Clarendon Press.

Atkinson, A. and Donev, A. (1996). Experimental designs optimally balanced for trend, *Technometrics* **38**: 333–341.

Box, G. and Draper, N. (1971). Factorial designs, the $|\mathbf{X}'\mathbf{X}|$ criterion and some related matters, *Technometrics* **13**: 731–742.

Cheng, C.-S. (1985). Run orders of factorial designs, *In* L. LeCam and R. Olshen (eds), *Proceedings of the Berkeley Conference in Honor of Jerzy Neyman and Jack Kiefer*, Wadsworth, pp. 619–633.

Cox, D. (1951). Some systematic experimental designs, *Biometrika* **38**: 312–323.

Daniel, C. and Wilcoxon, F. (1966). Fractional 2^{p-q} plans robust against linear and quadratic trends, *Technometrics* **8**: 259–278.

Donev, A. and Atkinson, A. (1988). An adjustment algorithm for the construction of exact $\mathcal{D}$-optimum experimental designs, *Technometrics* **30**: 429–433.

Farell, R., Kiefer, J. and Walbran, A. (1967). Optimum multivariate designs, *Proceedings of the 5h Berkeley Symposium*, University of California Press, Berkeley, pp. 113–138.

Freeny, A. and Lai, W. (1997). Planarization by chemical mechanical polishing: a rate and uniformity study, *In* V. Czitrom and P. Spagon (eds), *Statistical Case Studies for Industrial Process Improvement*, Philadelphia: ASA-SIAM, pp. 251–264.

Goos, P. (2001). *The optimal design of blocked and split-plot experiments*, Ph.D. dissertation, Katholieke Universiteit Leuven, Belgium.

Harville, D. (1997). *Matrix Algebra from a Statistician's Perspective*, New York: Springer-Verlag.

Joiner, B. and Campbell, C. (1976). Designing experiments when run order is important, *Technometrics* **18**: 249–259.

Tack, L. and Vandebroek, M. (2001). $(\mathcal{D}_t, C)$-optimal run orders, *Journal of Statistical Planning and Inference* **98**: 63–80.