A POLYTOPE METHOD FOR ESTIMATING FACTOR MAIN EFFECTS EFFICIENTLY

Bruce E. Ankenman

Dept. of Industrial Engineering and Management Sciences
Northwestern University
Evanston, IL. 60208, U.S.A.

Russell C.H. Cheng
Susan M. Lewis

School of Mathematics
University of Southampton
Southampton, SO17 1BJ, U.K.

ABSTRACT

Consider the problem of identifying important factors influencing a response in a simulation experiment where the number of factors is large. When the direction of the effect of factors is known, the method of sequential bifurcation is effective for quickly removing non-influential factors. Though good, the method is not fully efficient in that not all the information available is fully utilized. We present a method based on a polytope construction that makes use of all available information and which is therefore more efficient. In this paper we focus on the deterministic case to highlight its theoretical foundation. The method can however be extended to the stochastic case. Numerical examples are given comparing the new method with sequential bifurcation showing its improved performance.

1 INTRODUCTION

We consider the use of a linear model to represent the output of an simulation experiment where there are a large number of factors, but where many of the factors are expected to have little or no influence on the response output. In many discrete event simulations the direction of influence of each factor is known. For example, in a manufacturing process, two factors affecting the output rate of the process are the speed of operation of a machine and the number of machines available. Both these factors can be expected to increase output rate if they are increased. Thus the directions of their effects on the output of interest are known.

When the direction of influence of each factor is known, the method of sequential bifurcation is a simple but effective method of making simulation runs that enables elimination of non-influential factors not just one at a time but often in large groups simultaneously. The original method was proposed and studied in Kleijnen and Bettonvil (1997) for the deterministic case. Cheng (1997) gave a version for stochastic case. A more general version for the stochastic case has been given by Wan, Ankenman and Nelson (2006).

The use of sequential bifurcation can lead to significant speed up in identifying non-influential factors and more generally in estimating factor effects. However it is clearly not a fully efficient process in that factor effects are only initially determined to be either zero or non-zero. This is satisfactory for effects that are zero, as they are then completely identified and estimated. But non-zero effects are only estimated when they are completely isolated and not before. This is not usually fully efficient as combinations of observations may allow non-zero effects to be identified; a process not utilized in sequential bifurcation.

In this paper we make the process fully efficient, still by adding observations sequentially one at a time, but at each step we characterize the full space of possible factor coefficients that could give rise to the current set of observations. We do this by regarding each such feasible combination of factor coefficients, as being a vector point $\boldsymbol{\beta}$ in Euclidean space. Viewed this way, the set of all feasible factor coefficient combinations, turns out to be a convex polytope, that is, a region (possibly unbounded) in Euclidean space defined by a finite number of linear inequalities: $\mathbf{A}\boldsymbol{\beta} \geq \mathbf{d}$. The number of constraints increases as more observations are added, and the polytope shrinks. A reduction in the dimension of the polytope usually (but not always) corresponds to the identification of certain factor effect values. The polytope converges to a final degenerate zero-dimensional point at which stage all factor effects have been determined. The process is thus efficient in the sense that all the available information about possible factor coefficient values are characterised by the polytope.

However the rate of convergence is dependent on the choice of factor settings used in each succeeding simulation run. There are therefore interesting issues concerning the design of experiments, if factor effects are to be determined rapidly.

In the next section we set out the theory for the deterministic case. In Section 3 we describe the practical algorithm, and in Section 4 we give some numerical examples comparing our polytope method with sequential bifurcation. Section 5 sets out some conclusions and discusses some ongoing work to extend the method to the stochastic case.

2   THE LINEAR MODEL

Though it is not really necessary for the method to work, we shall for ease of exposition adopt a fairly explicit framework for our overall simulation experiment.

We suppose that the simulation comprises up to a maximum of k runs each providing an observation of the form

$$Y_i = \beta_0 + \sum_{j=1}^{k-1} \beta_j x_{ij}, \quad i = 1, 2, ..., k . \quad (1)$$

Here Y_i is the observed response in the ith run; x_j, $j = 1, 2, ..., k-1$ are $(k$-1) factors, with x_{ij} the level of factor j in the ith run; and β_j, $j = 0, 1, .., k-1$, are the unknown coefficient values that we wish to identify.

Our key assumption is that we know the direction of the each factor effect. We can therefore, if necessary with a change of sign to x_j, assume with no loss of generality that

$$\beta_j \geq 0, \quad j = 0, 1, .., k-1 . \quad (2)$$

We can write the system more compactly in matrix form as

$$\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} ,$$

where we follow standard design of experiments terminology and call $\mathbf{X}$ the *design matrix* with, as its first column, the vector with all entries equal to unity. We shall refer to the combination of factor levels used in a given run as a *design point*.

We adopt the sequential bifurcation assumption (which is not strictly necessary here) that each factor is applied at one of two levels scaled to be +1 and −1 .

We shall also assume that, were we to carry out the full set of k runs, we would definitely be able to identify all k of the β_j values. This is equivalent to assuming that $\mathbf{X}$ is non-singular with full rank k. The hope of course is that if we choose the right design we will be able to make use of the positivity assumption (2) and identify all the β_j values in considerably less than k runs.

Our method is sequential, so suppose that the experiment has reached the stage where observations have been taken at m design points and there are n coefficients yet to be estimated. Let the current design matrix corresponding to these m design points and n coefficients be

$$\mathbf{D} = [\mathbf{D}_1 \; \mathbf{D}_2]$$

where $\mathbf{D}_1$ is the square matrix formed from the first m rows and m columns. We assume that the columns are sorted so that $\mathbf{D}_1$ is nonsingular and so has inverse $\mathbf{D}_1^{-1}$. Let the true coefficients be $\boldsymbol{\beta}$ and the m observations to date be $\mathbf{y}$. For

the deterministic case that we are considering, the observations therefore have the form

$$\mathbf{y} = \mathbf{D}\boldsymbol{\beta} . \quad (3)$$

Our objective is to find $\boldsymbol{\beta}$.

The general solution of the equation (3) is well known (see Searle, 1971) to be expressible as:

$$\tilde{\boldsymbol{\beta}} = \mathbf{GD}\boldsymbol{\beta} + (\mathbf{I}_{n \times n} - \mathbf{GD})\mathbf{z} \quad (4)$$

where $\mathbf{G}$ is a generalized inverse of $\mathbf{D}$ and $\mathbf{z}$ is an n - dimensional column vector of arbitrary 'spanning' variables. A convenient generalized inverse $\mathbf{G}$ of $\mathbf{D}$ is:

$$\mathbf{G} = \begin{bmatrix} \mathbf{D}_1^{-1} \\ \mathbf{0}_{(n-m) \times m} \end{bmatrix} \begin{matrix} \}m \\ \}n-m \end{matrix} . \quad (5)$$

Taking $\mathbf{G}$ in the form (5) we have that $\mathbf{GD}$ takes the simple form:

$$\mathbf{GD} = \begin{bmatrix} \mathbf{D}_1^{-1} \\ \mathbf{0}_{(n-m) \times m} \end{bmatrix} [\mathbf{D}_1 \; \mathbf{D}_2] = \begin{bmatrix} \mathbf{I}_{m \times m} & \mathbf{D}_1^{-1}\mathbf{D}_2 \\ \mathbf{0}_{(n-m) \times m} & \mathbf{0}_{(n-m) \times (n-m)} \end{bmatrix} \quad (6)$$

(where $\mathbf{I}_{m \times m}$ is the $(m \times m)$ identity matrix).

Now replace $\mathbf{D}\boldsymbol{\beta}$ in (4) by $\mathbf{y}$ (from (3)), and use the fact that all components have to be positive by assumption. This yields the set of constraints that has to be satisfied by the spanning variables as:

$$\mathbf{Gy} + (\mathbf{I}_{n \times n} - \mathbf{GD})\mathbf{z} \geq \mathbf{0} . \quad (7)$$

Using (6) in (7) and writing

$$\mathbf{z} = \begin{pmatrix} \mathbf{u} \\ \mathbf{v} \end{pmatrix} \begin{matrix} \}m \\ \}n-m \end{matrix}$$

we have the general solution:

$$\tilde{\boldsymbol{\beta}} = \mathbf{Gy} + (\mathbf{I}_{n \times n} - \mathbf{GD})\mathbf{z}$$
$$= \mathbf{Gy} + \left(\mathbf{I}_{n \times n} - \begin{bmatrix} \mathbf{I}_{m \times m} & \mathbf{D}_1^{-1}\mathbf{D}_2 \\ \mathbf{0}_{(n-m) \times m} & \mathbf{0}_{(n-m) \times (n-m)} \end{bmatrix} \right) \begin{pmatrix} \mathbf{u} \\ \mathbf{v} \end{pmatrix} .$$

This simplifies to

$$\tilde{\boldsymbol{\beta}} = \begin{pmatrix} \mathbf{D}_1^{-1}\mathbf{y} - \mathbf{D}_1^{-1}\mathbf{D}_2\mathbf{v} \\ \mathbf{v} \end{pmatrix} \begin{matrix} \}m \\ \}n-m \end{matrix} . \quad (8)$$

Thus (7) reduces to

$$\begin{pmatrix} \mathbf{D}_1^{-1}\mathbf{y} - \mathbf{D}_1^{-1}\mathbf{D}_2\mathbf{v} \\ \mathbf{v} \end{pmatrix} \geq \mathbf{0} \qquad (9)$$

a system of n constraints involving just $(n - m)$ of spanning variables.

This system defines a polytope containing the unknown correct solution, $\boldsymbol{\beta}^*$ say, i.e.

$$\boldsymbol{\beta}^* = \begin{pmatrix} \mathbf{D}_1^{-1}\mathbf{y} - \mathbf{D}_1^{-1}\mathbf{D}_2\mathbf{v}^* \\ \mathbf{v}^* \end{pmatrix} \geq \mathbf{0}$$

for some unknown $\mathbf{v}^*$.

Observe now that we can simply test to see if any of β_i are fixed at a given value by the inequalities (9). Each such coefficient must then be at its correct value and so be determined. In particular any inequality which is *always active* will correspond to a *fixed zero* β_j which must therefore be at its correct null (i.e. zero) value.

This observation provides the means by which we can identify those coefficients that each additional observation fixes, taking into account all previous observations.

However we still have to come up with an efficient way of implementing this observation. We discuss this in the next section.

3 IDENTIFICATION OF COEFFICIENTS

There are various ways of determining which coefficients are fixed by (9). We present one method that we have found to be effective and reasonably efficient computationally.

We consider the constraints in the following form:

$$\begin{aligned} \mathbf{D}_1^{-1}\mathbf{y} - \mathbf{D}_1^{-1}\mathbf{D}_2\mathbf{v} \geq \mathbf{0} \\ \mathbf{v} \geq \mathbf{0} \end{aligned} . \qquad (10)$$

We now split the vector $\mathbf{D}_1^{-1}\mathbf{y}$ (which comprises known numeric values) into three subvectors

$$\mathbf{D}_1^{-1}\mathbf{y} = \begin{pmatrix} \boldsymbol{\theta}_p \\ \boldsymbol{\theta}_q \\ \boldsymbol{\theta}_r \end{pmatrix} \qquad (11)$$

where all components of $\boldsymbol{\theta}_p$ are strictly positive, all the components of $\boldsymbol{\theta}_q$ are exactly zero, and all the components of $\boldsymbol{\theta}_r$ are strictly negative. We use the subscript to indicate the dimension of the component. Thus for example we assume that there are p strictly positive components. (We

also have $p + q + r = m$.) The constraints therefore take the form

$$\begin{aligned} \boldsymbol{\theta}_p - \mathbf{A}_p\mathbf{v} \geq \mathbf{0}_p \\ \mathbf{A}_q\mathbf{v} \geq \mathbf{0}_q \\ \boldsymbol{\theta}_r - \mathbf{A}_r\mathbf{v} \geq \mathbf{0}_r \\ \mathbf{v} \geq \mathbf{0}_{n-m} \end{aligned} . \qquad (12)$$

where $\mathbf{A}_p, \mathbf{A}_q$ and $\mathbf{A}_r$ are the relevant submatrices of $\mathbf{D}_1^{-1}\mathbf{D}_2$ in (10).

To illustrate the discussion consider the 3-factor example where $\mathbf{X}$ is the design matrix

$$\mathbf{X} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & -1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 \end{bmatrix}, \qquad (13)$$

and suppose that $\beta_0 > 0$, $\beta_1 = \beta_2 = 0$, $\beta_3 > 0$ so that

$$y_i = \beta_0 + \beta_3 x_i.$$

At the stage when just two observations have been obtained, we have that

$$\mathbf{D}_1 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \text{ and } \mathbf{D}_2 = \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix},$$

$y_1 = \beta_0 + \beta_3$ and $y_2 = \beta_0 + \beta_3$. We find that (10) takes the form

$$\begin{aligned} \beta_0 + \beta_3 - v_2 \geq 0 \\ -v_1 \geq 0 \\ v_1 \geq 0 \\ v_2 \geq 0 \end{aligned}$$

In this case $\boldsymbol{\theta}_p - \mathbf{A}_p\mathbf{v} \geq \mathbf{0}_p$ in (12) reduces to the single equation $\beta_0 + \beta_3 - v_2 \geq 0$, whilst $\mathbf{A}_q\mathbf{v} \geq \mathbf{0}_q$ reduces to the single equation $-v_1 \geq 0$. There are no equations in the set $\boldsymbol{\theta}_r - \mathbf{A}_r\mathbf{v} \geq \mathbf{0}_r$, whilst $\mathbf{v} \geq \mathbf{0}_{n-m}$ reduces to $v_1 \geq 0$, $v_2 \geq 0$.

For the general case, note that if we drop some of the constraints from the set (12), this relaxes the set of constraints and so widens the set of possible solutions. Thus if we omit certain constraints and *still* find certain coefficients determined, then they must be at their correct value.

We focus first on those $\mathbf{v}$ which are always null by considering the reduced system

$$\begin{aligned} \mathbf{A}_q\mathbf{v} \geq \mathbf{0}_q \\ \mathbf{v} \geq \mathbf{0}_{n-m} \end{aligned} \qquad (14)$$

There is no point in including the constraints $\boldsymbol{\theta}_p - \mathbf{A}_p \mathbf{v} \geq \mathbf{0}_p$ in this check for null v's, because $\boldsymbol{\theta}_p > \mathbf{0}$, which implies that $\boldsymbol{\theta}_p - \mathbf{A}_p \mathbf{v} > \mathbf{0}_p$ for all sufficiently small $\mathbf{v}$. Thus these constraints are never active in such a check. Also we do not include $\boldsymbol{\theta}_r - \mathbf{A}_r \mathbf{v} \geq \mathbf{0}_r$ as $\boldsymbol{\theta}_r < \mathbf{0}$, and as we are only trying to detect nulls we will not be in a position to identify such a constraint as being always active with a fixed and positive left-hand side. These constraints wold be of interest if we were trying to identify non-nulls, but we are not doing so here. Their omission does not invalidate the overall process, but may make it less efficient. Based on numerical experiments, we conjecture that the loss of efficiency, if any, is not great.

Note also that in this test for null v's we may find that certain of the constraints $\mathbf{A}_q \mathbf{v} \geq \mathbf{0}_q$ are always active. For any such constraint this must mean that the corresponding component β_i is null as well.

This last possibility occurs in example (13), when (14) reduces to

$$-v_1 \geq 0$$
$$v_1 \geq 0 \ .$$
$$v_2 \geq 0$$

We see that the first two constraints are always active, showing that $\beta_1 = \beta_2 = 0$.

In general the method of Freund, Roundy and Todd (1985) can be used to find all the nulls with just one linear program. For convenience we set out their Proposition 1 here in a simplified form that is sufficient for our purpose.

Proposition. Consider the linear program:

$$\max_{\mathbf{x},\mathbf{y}} \mathbf{1}^{\mathrm{T}} \mathbf{y} \qquad (15a)$$

subject to

$$\mathbf{A}\mathbf{x} + \mathbf{y} \leq \mathbf{0}$$
$$\mathbf{0} \leq \mathbf{y} \leq \mathbf{1} \qquad (15b)$$

Then the set of always active constraints in the set of inequalities

$$\mathbf{A}\mathbf{x} \leq \mathbf{0}$$

has indices that comprise precisely the set

$$\{i \mid y_i^* = 0\}$$

where $(\mathbf{x}^*, \mathbf{y}^*)$ is any (optimal) solution of (15a and b). □

All nulls found in the above way can be suppressed from the constraints.

If no nulls are found, then the procedure stops (except that we can identify trivial cases $\tilde{\beta}_i = \theta_i - \mathbf{a}_i \mathbf{v}$, belonging to the first set of constraints $\tilde{\boldsymbol{\beta}}_p = \boldsymbol{\theta}_p - \mathbf{A}_p \mathbf{v} \geq \mathbf{0}_p$, where $\mathbf{a}_i = \mathbf{0}$, when β_i is obviously fixed at $\beta_i = \theta_i$). However if some nulls are found, then it may place sufficient restrictions to allow certain of the coefficients corresponding to constraints in the first set, $\tilde{\boldsymbol{\beta}}_p = \boldsymbol{\theta}_p - \mathbf{A}_p \mathbf{v} \geq \mathbf{0}_p$, to be identified as fixed. There seems no easy way of doing this with a single test. However each component β_i belonging to $\boldsymbol{\beta}_p$ can be tested separately as follows to see if it is fixed (and strictly positive).

Let β_i be one of the components belonging to $\boldsymbol{\beta}_p$. Then

$$\beta_i = \theta_i - \mathbf{a}_i \mathbf{v}$$

where $\theta_i > 0$. If therefore $\mathbf{a}_i \mathbf{v}$ is identically zero, we must have $\beta_i = \theta_i$. Let us therefore consider the following two sets of constraints separately:

$$
\begin{array}{ccc}
-\mathbf{a}_i \mathbf{v} \geq \mathbf{0} & & \mathbf{a}_i \mathbf{v} \geq \mathbf{0} \\
\mathbf{A}_q \mathbf{v} \geq \mathbf{0}_q & \text{and} & \mathbf{A}_q \mathbf{v} \geq \mathbf{0}_q \qquad (16) \\
\mathbf{v} \geq \mathbf{0}_{n-m} & & \mathbf{v} \geq \mathbf{0}_{n-m}
\end{array}
$$

If in both cases the first constraint is always active, then $\mathbf{a}_i \mathbf{v} = \mathbf{0}$ and so β_i must be fixed and equal to θ_i always, and thus equal to β_i^*, its true value. The linear programming method (15) can be used again for each of the two tests.

In the example (13) the two sets of constraints (16) can be set up for $i = 1$ only and we have simply

$$
\begin{array}{ccc}
-v_2 \geq 0 & & -v_2 \leq 0 \\
-v_1 \geq 0 & \text{and} & -v_1 \geq 0 \\
v_1 \geq 0 & & v_1 \geq 0 \\
v_2 \geq 0 & & v_2 \geq 0
\end{array}
$$

From the first set we see that the first inequality is always active as we must have both $-v_2 \geq 0$ and $v_2 \geq 0$. However the second set of inequalities only requires $v_2 \geq 0$ so that the first inequality in this second set is not always active. Thus β_0 is not yet identified, and because v_2 is not yet identified either, β_3 is not yet identified either. To identify these coefficients a further observation or observations are required. We will return to this example shortly.

In summary we see that our proposed method reduces to essentially two steps.

Step 1. Use (15) to identify which of the constraints in (14) are always active. Each always-active constraint corresponds to a null β_i. If there are no nulls the process

stops for this stage, except for trivial components of $\widetilde{\boldsymbol{\beta}}_p = \boldsymbol{\theta}_p - \mathbf{A}_p \mathbf{v} \geq \mathbf{0}_p$, where $\mathbf{a}_i = \mathbf{0}$; when β_i is obviously fixed at $\beta_i = \theta_i$.

Step 2. If there are nulls then, for each *i*, consider each one of the two sets of constraints in (16) separately, again using (15) to see which constraints are always active.. If the *first* constraint in the two sets are both always active then the corresponding β_i is fixed and positive.

These two steps are carried out as each new observation is obtained, until the values of all coefficients have been found.

A point to note is that once a coefficient has been identified, then that factor can be dropped from further consideration in the linear model. Moreover dropping factors in this way can mean that some of the design points already used may become linearly dependent. In this situation we can also drop any observation already obtained, but which corresponds to a design point that is linearly dependent on other points that have been already observed.

In example (13), with just two observations we were able to conclude that $\beta_1 = \beta_2 = 0$. When we add a third observation we can therefore at the same time drop the first and second factors from further consideration. The full design matrix with three observations

$$\mathbf{D} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & -1 & 1 \\ 1 & 1 & -1 & -1 \end{bmatrix}$$

can thus be reduced to

$$\mathbf{D} = \begin{bmatrix} 1 & * & * & 1 \\ 1 & * & * & 1 \\ 1 & * & * & -1 \end{bmatrix},$$

where the asterisks indicate the suppressed first and second factors. But clearly in this reduced form the first two rows are identical and so are linearly dependent. We can thus thus suppress the second row as well and work with the further reduced matrix:

$$\mathbf{D} = \begin{bmatrix} 1 & * & * & 1 \\ * & * & * & * \\ 1 & * & * & -1 \end{bmatrix}.$$

which is essentially a nonsingular (2×2) matrix. It is obvious in this case that the third observation together with the first observation and the fact that $\beta_1 = \beta_2 = 0$ will al-

low both β_0 and β_3 to be determined. Thus in this example all four coefficients have been identified with just three observations.

4 NUMERICAL EXAMPLES

We consider some numerical examples to illustrate the working of the polytope method and to compare it with sequential bifurcation. We shall consider four examples with 31 factors and one with 127.

As they are for illustration only and because we are considering just the deterministic case we can for simplicity take any non-zero coefficient to have value unity, all others being zero. Also, for the deterministic case the issue of sample size does not arise. For the more general case where the responses are observed with error, which we will address in a separate paper, a small pilot study would be required to gauge the magnitude of the observational error, so that sufficient observations are used at each design point to be able to detect coefficient values deemed to be practically significant.

We also need to select a suitable full design matrix $\mathbf{X}$ for the polytope method. In the stochastic case, the method has a clear advantage over sequential bifurcation in that we can employ a design with good characteristics. In our case we use orthogonal designs. Moreover we can take advantage of any prior information about the likely importance of certain coefficients by ranking them in their expected order of importance, and then also ranking the design points in lexicographic order to try to match the coefficient ordering. Thus, though of course in such a small example it makes little difference, the design matrix (13) should have rows re-ordered to be

$$\mathbf{X} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}.$$

The purpose of this reordering is to try to ensure that most of the null variables will correspond to higher numbered factors. The design will then identify these more quickly.

Note, as has been done in example (13), that it is best to start with all factors at their +1 settings. If then all the coefficients are zero, this one run will immediately detect them all as being zero. Moreover, even if not all zero, the value of the response in this first run will immediately set a common upper bound on all the coefficient values, as no β_i can then be greater than y_1. Though we have not made use of this fact, taking the first design point in this form ensures the polytope will always be a (bounded) polyhedron.

The entire general procedure described in this section has been implemented as an Excel VBA macro and tested on numerous examples. We give some examples to illustrate the method in operation and to compare it with sequential bifurcation. In the examples below SB refers to sequential bifurcation and PM to the polytope method.

Example 1 Here $k = 31$, with 4 non-null β's
$$\beta_0 = \beta_1 = \beta_3 = \beta_4 = 1$$
SB took 10 observations, PM took 9.

Example 2 Here $k = 31$, with 6 non-null β's
$$\beta_0 = \beta_1 = \beta_3 = \beta_4 = \beta_6 = \beta_8 = 1.$$
SB took 14 observations, PM took 12.

Example 3 Here $k = 31$, with 8 non-null β's
$$\beta_0 = \beta_1 = \beta_3 = \beta_4 = \beta_6 = \beta_8 = \beta_{21} = \beta_{27} = 1.$$
SB took 21 observations, PM took 17.

Example 4 Here $k = 31$, with 10 non-null β's
$$\beta_0 = \beta_1 = \beta_3 = \beta_4 = \beta_6 = \beta_8 = \beta_{17} = \beta_{19} = \beta_{21} = \beta_{27} = 1.$$
SB took 24 observations, PM took 18.

It will be seen from these examples that when the number of non-null coefficients is small (and ranged to one end in the order in which the coefficients appear) then there is little difference between the two methods, though the polytope method does do better. However as the number of non-null coefficients increases and as they become more spread in their positions in the list of all coefficients, then the difference widens with the polytope method becoming significantly more advantageous.

Figure 1 shows the sequence of observations and the resulting identification of coefficient values in the sequential bifurcation case for Example 4. Figure 2 shows the corresponding sequence for the same example using the polytope method.

Our fifth and last example is where there are 127 factors.

Example 5 Here $k = 127$, with 14 non-null β's
$$\beta_0 = \beta_1 = \beta_3 = \beta_4 = \beta_6 = \beta_8 = \beta_{34} = 1$$
$$\beta_{59} = \beta_{65} = \beta_{84} = \beta_{103} = \beta_{104} = \beta_{117} = \beta_{121} = 1$$
SB took 50 observations, PM took 37.

It will be seen that in this case the polytope method has performed significantly better.
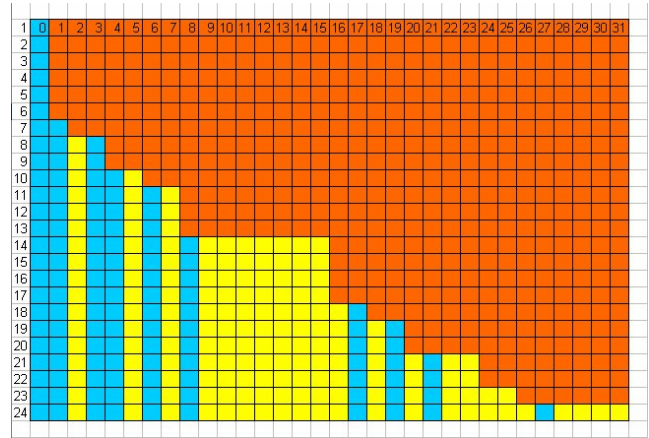

Figure 1: The Sequence of Coefficient Determinations for Example 4 Using the Sequential Bifurcation Method. (Reading from the top down, each row is an additional observation. Yellow – null coefficient identified, blue – non-null coefficient identified.)
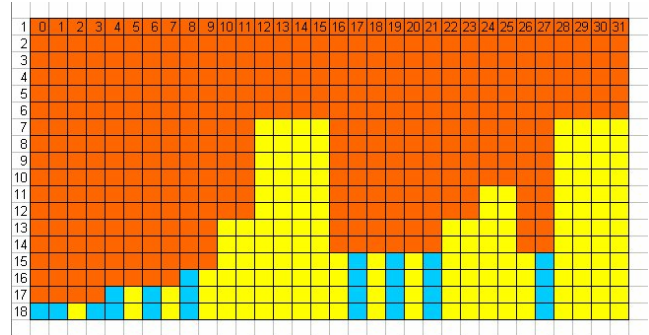

Figure 2: The Sequence of Coefficient Determinations for Example 4 Using the Polytope Method. (Reading from the top down, each row is an additional observation. Yellow – null coefficient identified, blue – non-null coefficient identified.)

5 CONCLUSIONS

We have presented a method for identifying the coefficients of a deterministic linear model that makes full use of available information. The theory indicates that the method will be superior to the sequential bifurcation method. The (admittedly limited) numerical comparisons indicate that the improvement is a worthwhile one.

Though we have only discussed the deterministic case the method is capable of extension to the stochastic case, and this will be the subject of a further article.

The examples presented in this article were calculated using VBA macros in an Excel Worksheet. The worksheet is available from the authors.

ACKNOWLEDGEMENT

REFERENCES

Bettonvil, B and J.P.C. Kleijnen. 1997. Searching for important factors in simulation models with many factors: Sequential Bifurcation. *European Journal of Operational Research*. 96 (1): 180-194.

Cheng, R.C.H. 1997. Searching for important factors: sequential bifurcation under uncertainty. In *Proceedings of the 1997 Winter Simulation Conference*, eds: S. Andradottir, K.J. Healy, D.H. Withers and B.L. Nelson. IEEE, Piscataway, 275-280.

Freund, R.M., R. Roundy, and M.J. Todd. 1985. Identifying the set of always active constraints in a system of linear inequalities by a single linear program. Sloan W.P. No. 1674-85 (Rev).

Searle, S.R. 1971. *Linear Models*. New York: Wiley.

Wan, H., Ankenman, B.E. and Nelson, B.L. 2006. Controlled sequential bifurcation: a new factor-screening method for discrete-event simulation. To appear in *Operations Research*.

AUTHOR BIOGRAPHIES

BRUCE E. ANKENMAN is an Associate Professor in the Department of Industrial Engineering and Management Sciences at the McCormick School of Engineering at Northwestern University. His current research interests include response surface methodology, design of experiments, robust design, experiments involving variance components and dispersion effects, and design for simulation experiments. He is a past chair of the Quality Statistics and Reliability Section of INFORMS, is an Associate Editor for *Naval Research Logistics* and is a Department Editor for *IIE Transactions: Quality and Reliability Engineering*.

RUSSELL C. H. CHENG is Professor and Head of the Operational Research Group at the University of Southampton. He has an M.A. and the Diploma in Mathematical Statistics from Cambridge University, England. He obtained his Ph.D. from Bath University. He is a former Chairman of the U.K. Simulation Society, a Fellow of the Royal Statistical Society and the British Computer Society, Member of the Operational Research Society. His research interests include: variance reduction methods and parametric estimation methods. He was a Joint Editor of the *IMA Journal of Management Mathematics*. His email and web addresses are <R.C.H.Cheng@maths.soton.ac.uk> and <www.maths.soton.ac.uk/staff/Cheng>.

SUSAN M. LEWIS holds a BSc and an MSc from London University and a PhD from the University of Southampton. She is Professor of Statistics at the University of Southampton and Director of the Southampton Statistical Sciences Research Institute where she leads the Design of Experiments Group (www.doe.soton.ac.uk). Her interests include factorial experiments and screening designs, designs for generalised linear models and algorithms for generating optimal designs. She was awarded the Royal Statistical Society's Greenfield Industrial Medal in 2005. She has served the Society as a Vice-President and as a Joint Editor of Series C (Applied Statistics) of the Society's journal. Her email and web addresses are <S.M.Lewis@maths.soton.ac.uk> and <www.maths.soton.ac.uk/staff/Lewis>.