LULEÅ UNIVERSITY OF TECHNOLOGY

1998:15

RESEARCH REPORT

A New Algorithm for Constructing Optimal Experimental Designs

JOHAN CARLSON
ANDERS GRENNBERG

November 1998

Department of Computer Science and Electrical Engineering
Division of Industrial Electronics

1998:15 • ISSN: 1402 – 1528 • ISRN: LTU - FR - - 98/15 - - SE

Abstract

In any experimental science we are sometimes confronted with new experimental situations, where the underlying physical models are unknown. After a first attempt at identifying physical variables that could effect the experimental result, we need to perform what is called a screening experiment. The objective is to conduct a small number of experiments that span as much of the experimental space as possible, to merit further experiments in smaller, more promising test systems.

In this report we present an algorithm that, given a set of possible experiments, selects a subset of these. The selection is designed to cover as much variation as possible in the experimental space, and to assure a minimized expected mean squared error in the model parameter estimation.

We verified the algorithm with an example taken from organic synthesis.

Contents

Chapter 1

Introduction

1.1 Background to the Problem

In almost any experimental science, planning the experiments is a difficult as well as an important step. First of all, the variables that are expected to affect the result must be identified. Usually, most of these variables can take on several different values, and as a consequence, there is a large number of possible experiments. Experiments can be both expensive and time-consuming, which makes it impossible or at least impractical to conduct them all. If this is the case, selecting a subset of these experiments becomes very important. There are several different ways of doing this selection, all depending on how much we know in advance about the examined system.

In this report we present a new method for selecting a number of experiments, without having to know anything about the effect of the variables.

The algorithm was developed as part of a project together with Rolf Carlson at Tromsø University, Norway. The goal of the project was to find a set of experiments for screening of discrete variations in organic synthesis (see also [1]). The methodology developed in this case could later be applied to our own project, regarding multiphase flow measurements.

1.2 Outline of the Report

In the next sections we will describe some of the principles of experimental design that exist today. At the end of this chapter we give a short introduction to the ideas of our new algorithm. Chapter 2 first gives a mathematical interpretation of the problem, followed by a detailed derivation of an optimality criterion and an algorithm to achieve it.

Chapter 3 contains two examples of applications of the algorithm, and some results verifying that the algorithm works. In the last chapter we discuss the result and how an algorithm of this kind could be used in practice.

1.3 Principles of Experimental Design

All experiments start with a design, not always a good one, but still some plan. Often the experimenter knows something about the system he or she wants to examine. Maybe the first experiment does not work as planned, which forces the experimenter to modify the conditions and try again. Doing this, and changing one of the experimental variables at a time, not only consumes time, but also risks that some effects are missed or misinterpreted. By changing one variable at a time, interactions between different variables will be missed, and it will be difficult to draw any conclusions about the examined system.

A better way of doing it is to first identify the variables that we believe affect the experimental result, select different values for them, and then perform experiments with all combinations of these variables. If this is done properly, we will be able to draw conclusions about the effect of individual variables as well as of interaction effects. However, the number of possible variables can be large, the space of possible experiments can have high dimension, which means that the geometric intuition of how to cover the volume of interest can be insufficient. For this, we need some selection algorithm.

The next two sections will give examples of some experimental design strategies, and how to interpret results. In the last section of this chapter we will give a brief introduction to the fundamental ideas behind our new method.

1.3.1 Factorial Designs

A useful class of experimental designs are factorial designs (see chapter 10 in [2]), and especially two-level factorial designs. To perform a general factorial design, the experimenter selects a number of levels for each of the variables to be examined, and then runs experiments for all possible combinations of these. In a two-level factorial design, the experimenter limits the number of levels for each variable to two, typically one high and one low level. This is a very powerful way to get information about the examined variables in relatively few experiments. This method also gives information about interaction effects between the variables. With limited knowledge about the system to be examined, two-level factorial designs are a powerful, as well as commonly used, way to perform a first examination of a new experimental situation.

However, even a two-level factorial design leads to a quite large number of experiments to perform, possibly a lot more than we can afford in a first, preliminary round. One solution to this problem is to use something called a *fractional factorial design* (see chapter 12 in [2]). If, for example, some of the higher order interaction effects can be neglected, the design matrix can be reduced to half. In this way, the redundancy in the full factorial matrix can be used to cut down on the number of experiments.

1.3.2 Other Design Methods

Two-level factorial design is very commonly used and is a very practical tool. However, the selection of variable levels is entirely up to the experimenter, and no optimality criterion is considered. Assume, for example, that the variables in a certain experiment can take on a large number of different values, typically a lot more than two. Selecting

levels for a two-level factorial design then becomes important, since performing a full factorial experiment is unfeasible. In general, experimentation is an iterative process, where the model has to be modified as we go along. It is therefore not very wise at the first steps, to perform a large number of experiments only in order to get a good estimate of the model parameters. Instead, it is a better idea to try to cover the experimental variation with as few experiments as possible. After that, the model can probably be modified, since we hopefully know more about the effects of the variables. Also, we will probably have gained some information about the region in which the variables are interesting. This will allow us to reduce the possible number of experiments even more, and after that, perform additional experiments to build a better model.

In the literature, several methods for constructing *optimal* designs have been discussed (see for example [3]), and with from that discussions follows a range of criteria. In the book [4], Fedorov discusses several optimality criteria and suggests algorithms to obtain the corresponding designs. One of these, the *D-optimality* criterion, was proposed in [1] for solving the selection problem described above. However, the proposed algorithm has some weaknesses and disadvantages that make it difficult to use in practice. The algorithm maximizes the determinant of a matrix $\mathbf{X}^T\mathbf{X}$, and the main problem is that in practice, $\mathbf{X}$ is often rank deficient, thus causing the determinant to be zero.

1.4 An Optimality Criterion for the New Method

In this report, we present an algorithm based on the *Minimum Mean Squared Error* (MMSE) criterion. Consider the linear model

$$\mathbf{X} \cdot \mathbf{a} = \mathbf{y} + \boldsymbol{\varepsilon} = \mathbf{f}. \tag{1.1}$$

The least squares solution of this is

$$\hat{\mathbf{a}} = \mathbf{X}^{\dagger}\mathbf{f}, \tag{1.2}$$

where $\mathbf{X}^{\dagger}$ is the pseudo-inverse of $\mathbf{X}$. If the error $\boldsymbol{\varepsilon}$ has zero mean, and $\mathbf{X}$ has full column rank, $\hat{\mathbf{a}}$ will be an unbiased and unique estimate of $\mathbf{a}$. If $\mathbf{X}$ does not have full column rank, there will be infinitely many solutions. In that case, using the pseudo-inverse will give us the shortest solution. Not having full rank suggests that the model contains dependent variables, but before performing any experiments, there is no information available to solve this problem. Later in the report we will show that the bias is independent of $\boldsymbol{\varepsilon}$.

In the special case when $\mathbf{X}$ has full column rank, equation (1.2) can be re-written as

$$\hat{\mathbf{a}} = \left(\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T\mathbf{f}. \tag{1.3}$$

The aim of our algorithm is to find the matrix $\mathbf{X}$ that minimizes the mean squared error of $\hat{\mathbf{a}}$, that is, to find the matrix $\mathbf{X}$ that minimizes

$$E\left\{\| \hat{\mathbf{a}} - \mathbf{a} \|^2\right\}. \tag{1.4}$$

What makes this difficult is that the rows in $\mathbf{X}$ can not be chosen arbitrarily, but must be taken from a discrete set of candidate rows, given by a candidate matrix $\mathbf{C}$.

Chapter 2

Derivation of the Algorithm

2.1 Derivation of a MMSE Optimality Criterion

In order to explain the algorithm, we first need to specify the design problem in mathematical terms. Suppose we are faced with a new experimental problem, where we want to model a result of the experiment m as a function, $f_m(x_1, x_2, \ldots, x_N)$ of a set of variables, $x_1, x_2, \ldots, x_N$. A common approach is to assume that, in some bounded region, the result can be modeled as

$$f_m(x_1, x_2, \ldots, x_N) = \sum_{k=1}^{N} \alpha_k x_k + \sum_{k=1}^{N} \sum_{\substack{l=1 \\ l \neq k}}^{N} \beta_{k,l} x_k x_l + \sum_{k=1}^{N} \gamma_k x_k^2 + \varepsilon_m, \tag{2.1}$$

where ε_m is some experimental and model error. Written in vector notation, this is

$$f_m = \begin{bmatrix} x_{m1} & x_{m2} & \cdots & x_{mN}^2 \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \gamma_N \end{bmatrix} = \mathbf{x}_m \mathbf{a}^T + \varepsilon_m. \tag{2.2}$$

Then, if we conduct M experiments with different values for the variables, we will get the result vector $\mathbf{f}(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N)$. This can be written in matrix notation as

$$\mathbf{f} = \mathbf{X} \cdot \mathbf{a} + \boldsymbol{\varepsilon} = \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_M \end{bmatrix} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1N}^2 \\ x_{21} & x_{22} & \cdots & x_{2N}^2 \\ \vdots & \ddots & & \vdots \\ x_{M1} & x_{M2} & \cdots & x_{MN}^2 \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \gamma_N \end{bmatrix} + \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_M \end{bmatrix}. \tag{2.3}$$

Now, the best estimate of $\mathbf{a}$, in the least-square sense, is

$$\widehat{\mathbf{a}} = \mathbf{X}^{\dagger} \mathbf{f}, \tag{2.4}$$

where $\mathbf{X}^{\dagger}$ is the pseudo-inverse of $\mathbf{X}$, as given by the *Singular Value Decomposition* (SVD) [5]. When $\mathbf{X}$ does not have full column rank, equation (2.3) will have infinitely

many solutions, and the pseudo-inverse will give the shortest of these. This means that the true value of $\mathbf{a}$ can be different from the mean of the estimate $E\{\hat{\mathbf{a}}\}$, i.e. $\hat{\mathbf{a}}$ can be a biased estimate of $\mathbf{a}$ in that case.

We now define the optimality criterion for the selection as

$$\mathbf{X}_p = \arg\min_{\mathbf{X}} E\left\{\| \hat{\mathbf{a}} - \mathbf{a} \|^2\right\}, \tag{2.5}$$

where $\hat{\mathbf{a}} = \mathbf{X}^\dagger \mathbf{f}$ and $\mathbf{X}_p$ are p rows selected among the rows of some candidate matrix $\mathbf{C}$, so that the mean squared error of the model parameters $\hat{\mathbf{a}}$ is minimized.

$$E\left\{\| \hat{\mathbf{a}} - \mathbf{a} \|^2\right\} = E\left\{(\hat{\mathbf{a}} - \mathbf{a})^T (\hat{\mathbf{a}} - \mathbf{a})\right\} \tag{2.6}$$

Inserting equation (2.4) into (2.6) gives

$$E\left\{\left(\mathbf{X}^\dagger \mathbf{f} - \mathbf{a}\right)^T \left(\mathbf{X}^\dagger \mathbf{f} - \mathbf{a}\right)\right\}. \tag{2.7}$$

Using equation (2.3), and the fact that $\boldsymbol{\varepsilon}$ is assumed independent of $\mathbf{X}$ and $E\{\boldsymbol{\varepsilon}\} = 0$, gives

$$\begin{aligned} & E\left\{\left(\mathbf{X}^\dagger \mathbf{X}\mathbf{a} + \mathbf{X}^\dagger \boldsymbol{\varepsilon} - \mathbf{a}\right)^T \left(\mathbf{X}^\dagger \mathbf{X}\mathbf{a} + \mathbf{X}^\dagger \boldsymbol{\varepsilon} - \mathbf{a}\right)\right\} \\ = \; & E\left\{\left(\mathbf{a}^T \mathbf{X}^T \mathbf{X}^{\dagger T} + \boldsymbol{\varepsilon}^T \mathbf{X}^{\dagger T} - \mathbf{a}^T\right)\left(\mathbf{X}^\dagger \mathbf{X}\mathbf{a} + \mathbf{X}^\dagger \boldsymbol{\varepsilon} - \mathbf{a}\right)\right\} \\ = \; & E\left\{\mathbf{a}^T \mathbf{X}^T \mathbf{X}^{\dagger T} \mathbf{X}^\dagger \mathbf{X}\mathbf{a} - \mathbf{a}^T \mathbf{X}^T \mathbf{X}^{\dagger T}\mathbf{a} + \boldsymbol{\varepsilon}^T \mathbf{X}^{\dagger T} \mathbf{X}^\dagger \boldsymbol{\varepsilon} - \mathbf{a}^T \mathbf{X}^\dagger \mathbf{X}\mathbf{a} + \mathbf{a}^T\mathbf{a}\right\}. \end{aligned} \tag{2.8}$$

Since $\mathbf{X}^\dagger \mathbf{X}$ is a projection matrix $\mathbf{P}_{\mathbf{X}}$ onto the row space of $\mathbf{X}$, (see [6], ch. 5), equation (2.8) simplifies to

$$\begin{aligned} & E\left\{\mathbf{a}^T \mathbf{P}_{\mathbf{X}} \mathbf{a} - \mathbf{a}^T \mathbf{P}_{\mathbf{X}} \mathbf{a} + \boldsymbol{\varepsilon}^T \mathbf{X}^{\dagger T} \mathbf{X}^\dagger \boldsymbol{\varepsilon} - \mathbf{a}^T \mathbf{P}_{\mathbf{X}} \mathbf{a} + \mathbf{a}^T \mathbf{a}\right\} \\ = \; & E\left\{\boldsymbol{\varepsilon}^T \mathbf{X}^{\dagger T} \mathbf{X}^\dagger \boldsymbol{\varepsilon} - \mathbf{a}^T \mathbf{P}_{\mathbf{X}} \mathbf{a} + \mathbf{a}^T \mathbf{a}\right\} \\ = \; & E\left\{\boldsymbol{\varepsilon}^T \mathbf{X}^{\dagger T} \mathbf{X}^\dagger \boldsymbol{\varepsilon} - \mathbf{a}^T \left(\mathbf{I} - \mathbf{P}_{\mathbf{X}}\right)\mathbf{a}\right\} = E\left\{\boldsymbol{\varepsilon}^T \mathbf{X}^{\dagger T} \mathbf{X}^\dagger \boldsymbol{\varepsilon} - \mathbf{a}^T \mathbf{P}_{\mathbf{X}}^\perp \mathbf{a}\right\} \\ = \; & E\left\{\boldsymbol{\varepsilon}^T \mathbf{X}^{\dagger T} \mathbf{X}^\dagger \boldsymbol{\varepsilon}\right\} + \| \mathbf{P}_{\mathbf{X}}^\perp \mathbf{a} \|^2, \end{aligned} \tag{2.9}$$

where $\mathbf{P}_{\mathbf{X}}^\perp$ is the projection matrix that projects on the orthogonal complement of the row space of $\mathbf{X}$. This term decreases when the rank of $\mathbf{X}$ increases. Thus, if $\mathbf{X}$ has full column rank, the second term in (2.9) vanishes and the error reduces to

$$J = E\left\{\boldsymbol{\varepsilon}^T \mathbf{X}^{\dagger T} \mathbf{X}^\dagger \boldsymbol{\varepsilon}\right\}. \tag{2.10}$$

This leads to the first step of our algorithm, where we select rows from a matrix of candidate experiments, so that the rows of the experimental matrix $\mathbf{X}$ are as orthogonal as possible. Doing this, $\mathbf{X}$ will have maximum possible rank, and the second term in (2.9) is minimized. The second step in the algorithm then appends $\mathbf{X}$ with rows that minimize (2.10). This will not alter the value of the second term in equation (2.9).

Since the expression inside the brackets is a scalar, this is equal to minimizing

$$J = E\left\{\mathrm{tr}\left(\boldsymbol{\varepsilon}^T \mathbf{X}^{\dagger T} \mathbf{X}^{\dagger} \boldsymbol{\varepsilon}\right)\right\}. \tag{2.11}$$

Noting that $\mathrm{tr}(\mathbf{AB}) = \mathrm{tr}(\mathbf{BA})$, equation (2.11) can be rewritten as

$$J = \mathrm{tr}\left(\mathbf{X}^{\dagger} \boldsymbol{\varepsilon}\boldsymbol{\varepsilon}^T \mathbf{X}^{\dagger T}\right). \tag{2.12}$$

We define the covariance matrix $\boldsymbol{\theta}$ as

$$\boldsymbol{\theta} = E\left\{\boldsymbol{\varepsilon}\boldsymbol{\varepsilon}^T\right\}, \tag{2.13}$$

which leads to

$$J = \mathrm{tr}\left(\mathbf{X}^{\dagger} \boldsymbol{\theta} \mathbf{X}^{\dagger T}\right). \tag{2.14}$$

When $\mathbf{X}$ is factored using the SVD, as

$$\mathbf{X} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T, \tag{2.15}$$

the pseudo-inverse $\mathbf{X}^{\dagger}$ is given by,

$$\mathbf{X}^{\dagger} = \mathbf{V}\boldsymbol{\Sigma}^{\dagger}\mathbf{U}^T, \tag{2.16}$$

where the diagonal of $\boldsymbol{\Sigma}^{\dagger}$ $(n \times m)$ contains the reciprocals $1/\sigma_1, \ldots, 1/\sigma_r$ of the non-zero diagonal of $\boldsymbol{\Sigma}$ $(m \times n)$ $\sigma_1, \ldots \sigma_r$. All other elements of $\boldsymbol{\Sigma}^{\dagger}$ are zero. Inserting this into (2.14), we obtain

$$J = \mathrm{tr}\left(\mathbf{V}\boldsymbol{\Sigma}^{\dagger}\mathbf{U}^T \boldsymbol{\theta} \mathbf{U}\boldsymbol{\Sigma}^{\dagger T}\mathbf{V}^T\right). \tag{2.17}$$

Considering the special case when $\boldsymbol{\theta} = \mathbf{I}\cdot\eta^2$, equation (2.14) reduces to

$$J = E\left\{\mathrm{tr}\left(\mathbf{V}\boldsymbol{\Sigma}^{\dagger}\boldsymbol{\Sigma}^{\dagger T}\mathbf{V}^T\right)\right\} \cdot \eta^2. \tag{2.18}$$

Since the trace operator is invariant to a change of basis, this is equivalent to

$$J = E\left\{\mathrm{tr}\left(\boldsymbol{\Sigma}^{\dagger}\boldsymbol{\Sigma}^{\dagger T}\right)\right\} \cdot \eta^2 = \left(\sum_{k=1}^{r} \frac{1}{\sigma_k^2}\right)\eta^2, \tag{2.19}$$

which leads to the criterion that the trace of $\boldsymbol{\Sigma}^{\dagger}\boldsymbol{\Sigma}^{\dagger T}$should be minimized, that is

$$\min_{\mathbf{X}} \sum_{k=1}^{r} \frac{1}{\sigma_k^2}, \tag{2.20}$$

where σ_k^2, $k = 1, 2, \ldots r$ are the eigenvalues of $\mathbf{X}^T\mathbf{X}$ and r is the rank of $\mathbf{X}$.

In the special case when $\mathbf{X}$ has full column rank, the ordinary least squares solution exists, and the pseudo-inverse is equal to

$$\mathbf{X}^{\dagger} = \left(\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T. \tag{2.21}$$

In section 2.3, we present an algorithm that first selects a matrix with as high rank as possible, and then includes additional rows to fulfill the criterion in equation (2.20). Note that the algorithm will work even if $\mathbf{X}$ does not have full rank. The model will still be reasonable, even though we have a residual in $(\widehat{\mathbf{a}} - \mathbf{a})$.

2.2 Constructing the Candidate Matrix

In the previous section we described how to estimate the model parameters given a model in the variables $x_1, x_2, \ldots, x_N$. Assume that the variable x_k can take on n_k different values, where n_k can be very large. This results in a total number of possible experiments M_{tot} as given in equation (2.22).

$$M_{tot} = \prod_{k=1}^{N} n_k. \tag{2.22}$$

A candidate matrix $\mathbf{C}$ containing all possible combinations of the variables can easily be constructed, and then appended with columns representing cross terms and quadratic terms according to the model in equation (2.1).

Depending on how much we know about the model and the variables, the values of x_k should be scaled. If we have no knowledge in advance about the variables, their maximum and minimum values should be scaled to, for example, 1 and -1 respectively. Alternatively, if we have information about the uncertainty in the variables, this can also be taken into account by scaling the values proportional to the variance of each variable.

When the candidate matrix $\mathbf{C}$ has been constructed, we are ready to determine the experimental matrix $\mathbf{X}$. The criteria we have now is that:

1. The rows of $\mathbf{X}$ should cover as much as possible of the variation in the experimental space, spanned by the rows in $\mathbf{C}$. That is, we should maximize the rank of $\mathbf{X}$.

2. The experimental matrix $\mathbf{X}$ should be well-conditioned, which, in this case, means that the quotient between the largest and the smallest eigenvalue of $\mathbf{X}^T\mathbf{X}$ should be close to one. This follows from the minimization of $\sum_{k=1}^{r} \frac{1}{\sigma_k^2}$.

In the next section we suggest an algorithm to do this selection

2.3 The Algorithm

In this section we will show how to use the SVD to select the experiments we want to perform. The algorithm consists of two steps. First we choose a number of rows equal to the rank of the candidate matrix. The second step then deals with selecting additional rows.

2.3.1 Step 1: Creating a Maximum Rank Matrix X

Any $m \times n$ matrix $\mathbf{C}$ can be factored into

$$\mathbf{C} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T, \tag{2.23}$$

where the columns of $\mathbf{U}$ are eigenvectors of $\mathbf{C}\mathbf{C}^T$, and the columns of $\mathbf{V}$ are eigenvectors of $\mathbf{C}^T\mathbf{C}$. If the rank of $\mathbf{C}$ is r, the r singular values on the diagonal of $\mathbf{\Sigma}$ are the square roots of the nonzero eigenvalues of $\mathbf{C}\mathbf{C}^T$ and $\mathbf{C}^T\mathbf{C}$. Furthermore, if $\mathbf{C}$ has rank r, the

first r columns of $\mathbf{V}$ form an orthogonal basis for the row space of $\mathbf{C}$, and the columns of $\mathbf{U}$ form an orthogonal basis for the column space of $\mathbf{C}$. Another useful property of the columns in $\mathbf{V}$ is that the first one points out the direction in which we have the largest variation in the row space, the next column points out the second most important direction, *etc.* If we, using as few experiments as possible, would like to span the row space of $\mathbf{C}$, and at the same time cover as much of the total variation as possible, selecting the experiments as the r first columns of $\mathbf{V}$ is optimal. In a real-life situation, however, it is not likely that we can set up experiments that satisfy this. What we have to do instead, is to select the row in $\mathbf{C}$ that is the most parallel to the first column in $\mathbf{V}$. That is, choose the row $\mathbf{c}_i^T$ in $\mathbf{C}$ that maximizes the absolute value of the scalar product $\mathbf{c}_i^T\mathbf{v}_1$.

When the first experiment has been chosen, we can remove the component in this direction from all rows in $\mathbf{C}$. The resulting matrix will have rank exactly one less than $\mathbf{C}$. Removing the projection onto row $\mathbf{c}_i^T$ from all other rows $\mathbf{c}_k^T$ is done according to equation (2.24)

$$\widehat{\mathbf{c}}_k = \mathbf{c}_k - \frac{\left(\mathbf{c}_i^T\mathbf{c}_k\right)}{\mathbf{c}_i^T\mathbf{c}_i}\mathbf{c}_i. \tag{2.24}$$

Below, we explain how this procedure can then be repeated to select as many rows as the rank r of the candidate matrix. The algorithm can be summarized in the following steps:

1. Let $\mathbf{C}_0$ be the original candidate matrix, consisting of all possible experiments (with scaled variables).
2. Factor $\mathbf{C}_j$ (where $\mathbf{C}_j$ is the candidate matrix after j iterations) in $\mathbf{U}, \mathbf{\Sigma}$, and $\mathbf{V}$ using *Singular Value Decomposition* (SVD).
3. Select the row $\mathbf{c}_i^T$ from $\mathbf{C}_j$ that is most parallel to the first column in $\mathbf{V}$, by finding the row that maximizes $\text{abs}\left(\mathbf{c}_i^T\mathbf{v}_1\right)$. Add the corresponding row of $\mathbf{C}_0$ to the experimental matrix $\mathbf{X}$.
4. Construct a new matrix $\mathbf{C}_{j+1}$ where the rows $\widehat{\mathbf{c}}_k^T$ are defined by equation (2.24).
5. Repeat from step 2 until the desired number of experiments has been selected (less than[1] or equal to the rank of $\mathbf{C}_0$).

2.3.2 Step 2: Adding Rows to the Design

Now, when we have a maximum rank matrix $\mathbf{X}$, and we want to perform some additional experiments, we must determine how to choose them. From the previous chapter, we know that we should construct $\mathbf{X}$ according to

$$\mathbf{X}_{N+1} = \arg\min_{\mathbf{X}} \sum_{k=1}^{r} \frac{1}{\sigma_k^2}, \tag{2.25}$$

[1]Even though it is possible to select fewer experiments than the rank of $\mathbf{C}$, and use the pseudo-inverse instead, it will not span the experimental space, which was what we wanted.

where the matrix $\mathbf{X}_{N+1} = \left[\begin{array}{c|c} \mathbf{X}_N^T & \mathbf{c}_k^T \end{array} \right]^T$ is the resulting matrix after including one more row $\mathbf{c}_k^T$, selected from one of the remaining candidate rows in $\mathbf{C}$.

Appending the design is thus done by including the row that decreases the value of (2.25) the most. This requires testing all possible rows in the candidate matrix that have not already been selected.

Testing all possible candidate rows is very time-consuming, and for a large candidate matrix it could be difficult to do the necessary calculations. We therefore suggest a much faster, sub-optimal alternative. In some cases this method could give a good enough approximation to the optimal selection. The first step in the algorithm is the same as before. When including additional rows, however, instead of testing all possible candidates, we select the row that increases the smallest singular value of $\mathbf{X}$ the most. This is achieved by selecting the row that has the largest component in the direction of $\mathbf{v}_r^T$, that is, the singular vector corresponding to the smallest singular value. The reason for this is that no singular value can decrease if a matrix is appended with an additional row [7], and the sum in equation (2.25) will be smallest if all σ_k^2 are equal. The implementation of this is very similar to the implementation of the first step.

Figure 2.1 compares the difference in performance between the two methods in a case where the candidate matrix was a 162×44 matrix with rank 35. This candidate matrix was taken from the experiments described in section 3.2. In this example, $\mathbf{X}$ does not

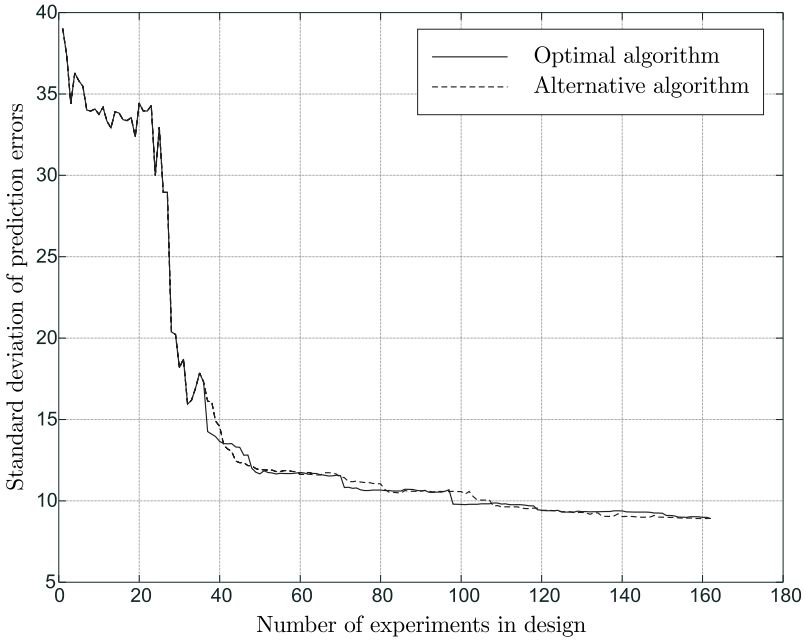

Figure 2.1: Performance of the optimal algorithm compared with the faster sub-optimal alternative. In this example the candidate matrix is 162×44 and has rank 35. The standard deviation of the prediction errors was used as a quality measure.

have full rank, and therefore the estimate of the model parameters $\hat{\mathbf{a}}$ might be biased. In this case, the ordinary least squares solution does not exist, since $\left(\mathbf{X}^T\mathbf{X}\right)^{-1}$ does not exist. For this example, the algorithm previously presented in [1] will not work.

Chapter 3

Application Examples

In this chapter we will give two examples on how this method can be used to generate experimental designs. The first is our own research project, the development of a non-invasive technique for measuring multiphase flow with pulsed ultrasound. The second example is the one that actually motivated the development of the algorithm.

3.1 Ultrasound Measurements of Multiphase Flows

In chemical and process industry, for example paper pulp industry and mining industry, multiphase flows play an important role. The long-term goal of this project is to develop a non-invasive technique for measuring mass fractions and mass fraction velocities in flows consisting of water and solid particles.

When transmitting an ultrasonic pulse through a flowing medium of this kind, there are a large number of factors affecting the waveform and energy of the received pulse. In [8] we found that, for example, particle concentration, particle size, temperature, and transmitter frequency affect the received pulse.

3.1.1 Experimental Design

For the current experimental setup, we have identified the following potential factors that we can vary: particle concentration (4 levels), particle sizes (2 levels), flow speed (10 levels), transmitter frequency (3 levels), and receiver position (9 levels). Even though some limitations have been introduced already, by setting the different possible levels of the variables, performing all of the possible experiments would result in a total of $4 \cdot 2 \cdot 8 \cdot 3 \cdot 3 = 576$ experiments. This is practically unfeasible, and some pre-selection must be done. Using our algorithm to rank the experiments allows us to select a subset of the experiments.

3.1.2 Results

Given the experimental setup from the previous section, we can construct a candidate as follows. For each of the variables, define the levels equally spaced in the interval $-1..1$. For example, a three-level variable will have the values $-1, 0, 1$. Running the algorithm

and plotting the value of (2.25) as a function of the number of experiments, we obtain results in figure 3.1.

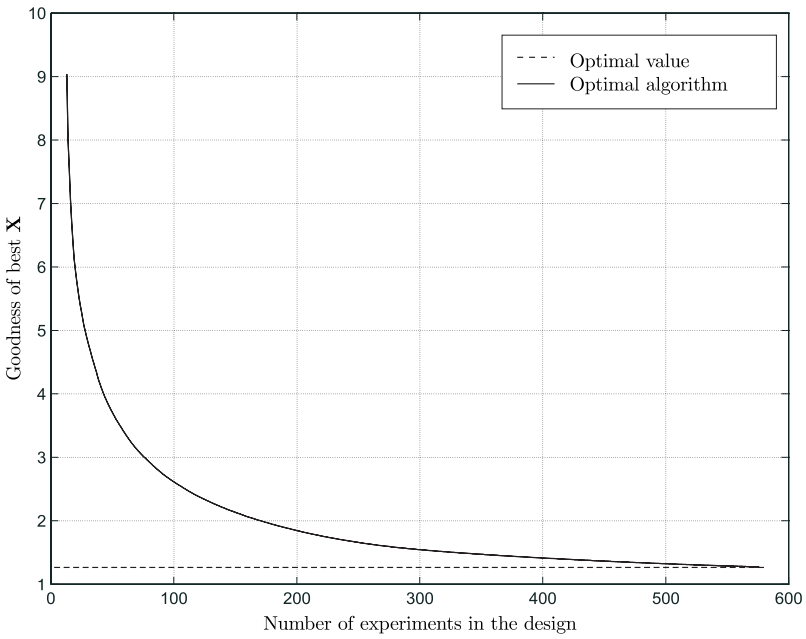

Figure 3.1: Performance of the algorithm when the candidate matrix is 576×15 with rank 13. The goodness of the selection is evaluated using equation (2.25). We see that the convergence is fast, and selecting between 50 and 100 rows gives a good approximation of the optimal result.

The next step will now be to perform the desired number of experiments. We can see from the figure that around 100 experiments could be sufficient to have a good estimate of the model parameters. At the time of writing, no results of these experiments are available, but will be reported later.

3.2 Screening of Variations in Organic Synthesis

An important field of research in organic synthesis is the development of efficient procedures for carrying out specific transformations. One of the problems encountered in this process is to determine which combination of reagent(s), catalyst(s), solvent(s), *etc.* will give the most promising result and thus merit further exploration, that is, to perform a screening of the experimental variations. These constituents can then be characterized by their *principal properties* (see [9] for a thorough description of this concept). Given these properties, we can construct a model that expresses the experimental result (for example the yield of a desired product) in terms of the principal properties. A common model is the one described in equation (2.1), where the variables x_i are the principal properties.

Even for a fairly simple synthesis, the number of possible experiments is very large, and we have to construct an experimental design that, with as few experiments as possi-

ble, covers the experimental variation. In the next section we compare our new selection algorithm with the widely accepted two-level factorial design.

3.2.1 Experimental Design

As an example of an organic synthesis problem, we have used *the indole experiment* presented in [10]. In this experiment, there were 8 variables, and a total of 162 candidate experiments. With a model including linear, cross product, and quadratic terms, the candidate matrix was 162×44, with rank 35. We compare observed experimental yield with predicted yield for designs selected using our algorithm with the result of performing all possible experiments. The goodness of the prediction depends not only on the selection of experiments, but also on the underlying model used for the prediction. In this case, the model in equation (2.1) was used, and the coefficients were estimated using *Partial Least Squares*, PLS (see [9], ch. 17).

In the next section we present the prediction results after selecting 50 experiments out of 162 possible.

3.2.2 Results of the New Algorithm

When the selected experiments were performed, The model parameters were estimated using (2.4). The resulting predictions and prediction errors can then be compared with the case where all experiments were conducted. In figure, 3.2, we see the prediction results when all possible experiments were performed.

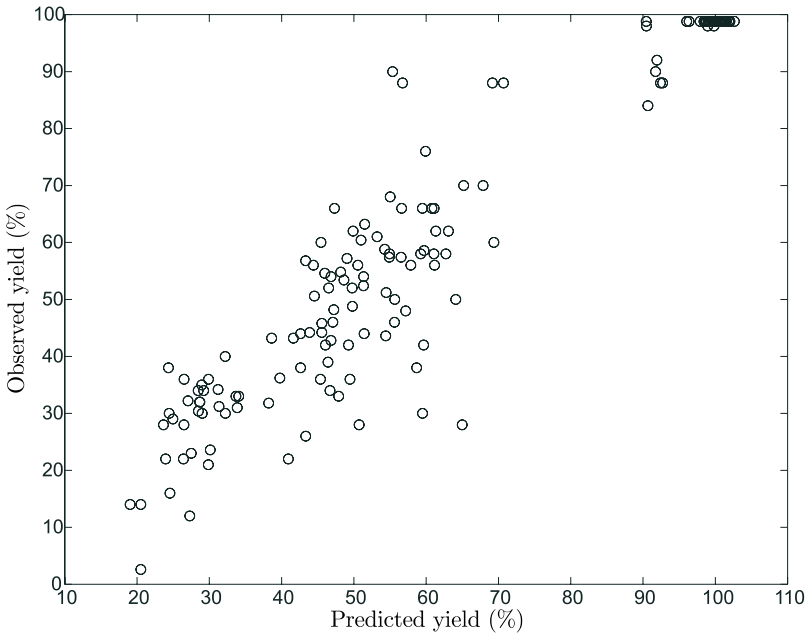

Figure 3.2: Predicted versus observed experimental yield in the Indole experiment when the model parameters were predicted using all of the 162 experiments.

In figure 3.3, we see the prediction results after performing only 50 experiments, all selected using the algorithm presented in the previous chapters.

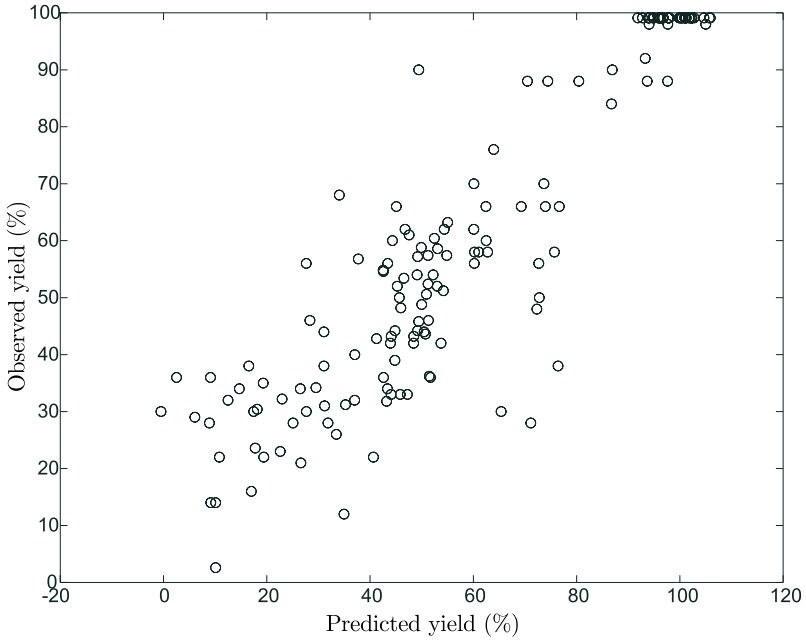

Figure 3.3: Predicted versus observed experimental yield when the model parameters where predicted using only 50 of the 162 experiments, selected using the algorithm presented in the previous chapters.

Chapter 4

Conclusions

From the organic synthesis example, where data after performing all possible experiments were available, we see that in that particular case, our selection algorithm works very well. In this case, we had the possibility to compare our results with the optimal case, and the results were very convincing. In our own research project, multiphase flow measurements, we have no experimental results yet that can verify the algorithm. This will be the next thing to do, and the results from this will be the next thing to publish.

Even though selection of experiments can be done with an algorithm that fulfills an optimality criterion like the one we presented in this report, one must not forget to experiment. Even if the selection works in an optimal way, we have to verify and refine the model by performing experiments. In a case like ours, where we have very little knowledge of the underlying physical model and we want a first, preliminary screening of the experimental space, an algorithm like this is probably a better approach than pure guessing. However, any pre-knowledge about the experimental situation must be taken into account in some way. In the multiphase flow case, this has been done in the selection of levels for the different variables. We know that certain transmitter frequencies, and certain particle sizes, are more interesting than others.

In the solution of a least squares problem, $\mathbf{Xa} = \mathbf{f}$, one assumption is that the errors are confined to the observations $\mathbf{f}$. If this is the case, the pseudo-inverse or the ordinary least squares solution works fine. If, however, we have errors in the model matrix $\mathbf{X}$, using *Total Least Squares* (TLS), as discussed in chapter 12 in [6], is a more natural approach. This can only be done after the experiments have been made.

Bibliography

[1] R. Carlson and J. Carlson, "Strategy for screening discrete variations in organic synthesis," in *Proc. Chimiometrie 97*, (Lyon, France), pp. C11–1 – C11–12, Societe de Chimie Industrielle, Dec. 1997.

[2] G. E. P. Box, W. G. Hunter, and J. S. Hunter, *Statistics for Experimenters*. John Wiley and Sons Inc., 1978.

[3] G. E. P. Box and N. R. Draper, *Empirical Model-Building and Response Surfaces*. John Wiley and Sons Inc., 1987.

[4] V. Fedorov, *Theory of Optimal Experiments*. Academic Press, 1972.

[5] G. Strang, *Linear Algebra and Its Applications*. Academic Press, 2nd ed., 1980.

[6] G. H. Golub and C. F. Van Loan, *Matrix Computations*. The Johns Hopkins University Press, 3rd ed., 1996.

[7] Å. Björck, *Numerical Methods For Least Squares Problems*. SIAM, 1996.

[8] J. Carlson, A. Grennberg, and J. Delsing, "Multiphase flow characterized by scattering of ultrasound," in *Proc. of the 9th Int. Conf. on Flow Measurement, FLOMEKO'98*, (Lund, Sweden), pp. 493–497, IMEKO, 1998.

[9] R. Carlson, *Design and Optimization in Organic Synthesis*. Elsevier, 1st ed., 1992.

[10] M. P. Prochazka and R. Carlson, "On the roles of lewis acid catalysts and solvents in the fischer indole synthesis," *Acta Chem. Scand.*, vol. 43, no. 5, pp. 651–659, 1989.