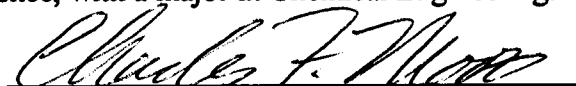


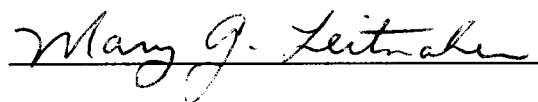
To the Graduate Council:

I am submitting herewith a thesis written by Brett Darnell entitled "Preprocessing Batch Profiles for Statistical Process Control." I have examined the final copy of this thesis for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Master of Science, with a major in Chemical Engineering.


Charles F. Moore, Major Professor

We have read this thesis
and recommend its acceptance:





Accepted for the Council:


Associate Vice Chancellor and
Dean of The Graduate School

Preprocessing Batch Profiles for Statistical Process Control

A Thesis
Presented for the
Master of Science
Degree
The University of Tennessee, Knoxville

Brett Darnell
December 1996

Abstract

Batch processes can be monitored and analyzed by applying SPC techniques. The primary motivation behind their use is to identify and study process variation in an effort to improve process efficiency and the quality and consistency of a product. Conventional SPC techniques are quite useful in identifying process variation, but they alone provide little indication as to its type or source.

In this study, a technique is developed for analyzing batch profiles. It is based on the same principles as a conventional SPC analysis, but it is capable of providing more quantitative as well as qualitative information. It uses a preprocessing step to analyze batch profiles before applying any SPC techniques. The preprocessing step is made up of a least square analysis that provides deterministic clues as to the source of variation. This analysis is capable of quantifying the effects of variation on a profile's initial condition, magnitude, and speed of development. It produces an adjusted profile that can be used to increase the sensitivity of an SPC analysis, and also generates a set of parameters that can be used in correlation studies and easily stored in a historical database.

Table of Contents

Chapter	Page
1. Introduction	1
1.1 Introduction to Batch Processes	1
1.2 Introduction to Batch Profiles.....	2
1.3 The Application of SPC to Batch Processes.....	4
1.4 Overview of the Technique Developed in this Study	4
1.5 Organization of Thesis.....	5
2. Literature Review.....	6
2.1 Management of Variation	6
2.2 SPC Analysis of Batch Processes.....	7
2.2.1 Control Charts for Discrete Measurements.....	8
2.2.2 Control Charts for Profiles	9
2.3 Multivariate Techniques Used to Monitor Batch Processes.....	12
2.4 Method of Least Squares.....	14
2.4.1 Linear Least Squares.....	15
2.4.2 Nonlinear Least Squares	17
3. Least Square Profile Analysis	18
3.1 Summary of Technique.....	18
3.2 Procedure for Analyzing Profiles	20
3.3 Procedure for Creating Control Charts	20
3.3.1 Determining a Reference Profile.....	20
3.3.2 Projection Control Charts	23
3.3.3 Calculating Control Limits for the Projection Control Chart ...	27
3.3.4 Parameter Control Charts.....	29

Chapter	Page
3.4 Dealing with Time During a Least Squares Analysis	32
3.5 Comments on Determining a Reference Profile	34
4. Examples of a Profile Analysis.....	40
4.1 Analysis of Profile 1	41
4.2 Analysis of Profile 2	45
4.3 Analysis of Profile 3	45
4.4 Analysis of Profile 4	51
5. Conclusions and Recommendations	56
5.1 Conclusions	56
5.2 Recommendations	57
Bibliography	59
Appendices	62
A. Matlab Programs Based on a Brute Force Search.....	63
B. Matlab Programs Based on a Mutivariable Search.....	70
C. Matlab Programs Based on a Single Variable Search.....	79
Vita	88

List of Figures

Figure	Page
3.1 Effects of Variation on Profiles.....	19
3.2 Flowsheet of the Procedure	21
3.3 Examples of the Use of Extrapolated Data	24
3.4 Two Types of Control Charts for Projections	30
3.5 Parameter Control Chart	31
3.6 Unsuccessful Projection without a Time (Speed) Adjustment.....	33
3.7 Unsuccessful Projection with a Time (Speed) Adjustment.....	35
3.8 Successful Projection with a Time (Speed) Adjustment.....	36
3.9 Collection of Profiles Used to Determine a Reference Profile.....	37
3.10 An Actual Profile with an Averaged Reference Profile.....	39
4.1 Profile Used for the Analysis	40
4.2 Plot of Profile 1 with the Reference Profile.....	42
4.3 Control Charts for Profile 1	43
4.4 Parameter Control Chart for Profile 1.....	44
4.5 Plot of Profile 2 with the Reference Profile.....	46
4.6 Control Charts for Profile 2.....	47
4.7 Parameter Control Chart for Profile 2.....	48
4.8 Plot of Profile 3 with the Reference Profile.....	49
4.9 Control Charts for Profile 3.....	50

Figure	Page
4.10 Parameter Control Chart for Profile 3.....	52
4.11 Plot of Profile 4 with the Reference Profile.....	53
4.12 Control Charts for Profile 4.....	54
4.13 Parameter Control Chart for Profile 4.....	55

Chapter 1

Introduction

1.1 Introduction to Batch Processes

A batch process consists of a series of operations that occur in a specified order, and produces products in discrete quantities. Typically, batch processes are transient in nature and do not operate at steady state conditions. Each batch process has a procedure called a recipe that relates all the information needed to successfully complete a batch. The recipe can be broken down into phases which are then described by their sequential steps. By changing the recipe of a batch, a batch process is capable of producing different grades of a product, or a different product altogether. Because batch processes usually consists of a multitude of operations, the proper scheduling of production activities is very important. By effective scheduling, production can be optimized by; reducing downtime between and during batches, minimizing inventories, and maximizing equipment usage and product throughput.

Typically, batch processes are used “ when product quantities are small, reaction times are long, feedstock supplies and market demands are uncertain, manufacturing procedures are likely to change, or a large variety of small volume products will be produced” as quoted from Fisher (1990). They are often used in the specialty

chemical, polymer, and pharmaceutical industries. Batch characteristics can also be found in continuous processes during start up, shut down, and grade changes.

A simple and straightforward example of a batch process is baking a cake. Although most industrial processes are more complex than cake baking, they do share many common characteristics. For instance, a recipe is used to specify all the information necessary to successfully bake a particular type of cake. Some examples of the information that might be contained in the recipe are: the ingredients required, the quantity of each ingredient, the steps to be performed, and the baking temperature and duration. Upon the completion of the recipe, a single cake is produced. To make additional cakes with the same texture, smell, and taste, the recipe and baking conditions must be repeated as closely as possible.

1.2 Introduction to Batch Profiles

In the control of a continuous process, the typical goal is to maintain key process variables at desired setpoints. Batch processes, however, do not operate at steady state conditions. The state of a batch process, by nature, is constantly changing. Therefore, most of the variables in a batch process change as the batch progresses. The measurement vs. time curves produced by these variables represent batch profiles.

Profiles may be generated for an entire batch or individual phases of a batch. Some typical batch profiles are temperature, pressure, composition, conversion, flowrates, and etc. Such profiles contain information about the state of a batch and the development of that state. Profiles can be used in the monitoring and control of batch processes in three ways:

Within-batch control. For most processes, target profiles have been established that define the trajectory variables should follow as a batch progresses. The development of a profile can then be compared to that target profile, with feedback adjustments made within the batch, based on the profile's deviation from the target.

Batch-to-batch control. At the end of each batch, a profile can be compared to a target or reference profile. Based upon the results of this comparison, feedback adjustments are made in the production of the following batch.

Continual process improvement. Profiles also can be used to gain a better understanding of the process. This understanding can then be used to make informed decisions about how the process and/or operations should be modified for improvements.

In conclusion, profiles can be a valuable tool in analyzing a batch process. They can provide important information about a process and can be used for process control, monitoring, and improvement.

1.3 The Application of SPC to Batch Processes

SPC techniques have been used in an effort to increase the effectiveness of monitoring batch processes. Typically, they are applied through the use of Shewhart charts and are used to monitor a single measurement made at the end of a batch. In an effort to better monitor and understand batch processes, SPC techniques have also been applied to batch profiles. But because batch profiles lack a fixed mean, are time dependent, and can contain autocorrelated data, the application and interpretation of SPC techniques to profiles is not as straightforward as with single measurement control charts. Some of the complications of applying control charts to profiles are further discussed in Chapter 3 of this thesis.

1.4 Overview of the Technique Developed in this Study

In this study, a technique is developed to analyze variation in batch profiles beyond a conventional SPC analysis. It attempts to establish a framework that can be used to identify possible sources of process variability. It uses a least square analysis to preprocess profiles which provides parametric quantification of certain forms of variation and an increase in SPC sensitivity. The purpose behind this study is to

develop new ways of monitoring and analyzing batch processes. It is motivated by the relatively limited number of techniques available for analyzing batch processes and the need for more analytical tools to aid in the study of process variation.

1.5 Organization of Thesis

A review of relevant literature is provided in Chapter 2. Several topics are discussed such as: the management of variation, the application of SPC techniques to batch processes, the use of multivariate techniques, and a description of least square projections. Then, the procedures used to create control charts and analyze profiles are presented in Chapter 3. An illustration of how the technique is used is provided in Chapter 4, followed by the recommendations and conclusions in Chapter 5.

Chapter 2

Literature Review

2.1 Management of Variation

As discussed by Moore (1991), there are many possible sources of variation in an industrial process. Poor operating procedures, operator shift changes, changes in material and energy balances, and equipment malfunctions are just a few examples of how variation may enter a process. The effects of variation may be seen in reductions in product quality, higher production cost, and/or low production rates. The incentives to identify and effectively manage variation are strongly connected to competitiveness and profitability. Moore has suggested, the management of variation can be divided into both a short term and long term approach.

The short term approach is focused on the transformation of variation. This is when variation is moved from a “critical process variable to a more benign location”. Such “transformations of variation” are typical of feedback and feedforward control systems. This approach may be effective in reducing the effects of variation in the product, but does not remove the variation from the process. Therefore, production efficiencies and costs are still likely to be affected.

Reduction or elimination of variation requires a more long term approach. This typically involves extensive studies of the process over long periods of time. These studies are used to identify the sources of variation and to gain a better understanding of the process itself. Once the sources of variation have been identified, then appropriate actions can be taken for their elimination. These actions are usually in the form of process modifications.

As discussed by Moore (1991), both approaches are important aspects of managing variation. It is highly unlikely that all the variation in a process can be adequately managed by applying only one approach. Therefore, a successful program for managing variation should incorporate both approaches based on engineering judgment and process knowledge.

2.2 SPC Analysis of Batch Processes

As stated by Marsh and Tucker (1991), “the primary goal of statistical process control is to detect process (and, in turn, product) variation”. This can be accomplished by developing control charts to monitor both the process and the product. These control charts can then be used to indicate when variation in the process occurs and when corrective action should be taken. The end result is hoped to be a reduction in the amount of batch-to-batch variation and the production of a consistent product. Marsh and Tucker (1991) have identified two ways of applying SPC control charts to batch

processes. The control charts discussed are based on discrete measurements of variables or product quality characteristics and on variable profiles generated during a batch.

2.2.1 Control Charts for Discrete Measurements

The control charts based on discrete measurements are typically used to detect batch-to-batch variation. The measurements are usually taken at the end of a phase or batch, and may consist of a single observation (sample) or of a group of observations. In many cases, the measurement is used to test a quality characteristic of the product and is made at the end of the batch. Marsh and Tucker (1991) have identified two types of control charts that can be created from such measurements. They are the $\bar{X}$ -R and the $\bar{X}$ -S charts. In order to construct these charts, multiple observations (samples) of a measurement must be made for each batch. An example might be a molecular weight distribution measurement made at the end of a polymeric batch reaction. For each batch, there are several samples taken and tested for the molecular weight distribution. The $\bar{X}$ -R chart is then constructed by calculating the average and range of the samples for each batch used in creating the control chart. The range of the samples is defined as the difference between the highest and lowest value. Once the average and range for each batch have been determined, the overall average and range for all the batches are calculated. These then become the centerlines on the control charts. Based on the overall average and range, the control limits for the $\bar{X}$ -

and R charts can be calculated. The Xbar-S control chart is created in the same manner, except the standard deviation of the samples are used instead of the range.

Marsh and Tucker state that when using control charts in this manner, the charts primarily indicate two things. First, the Xbar chart shows the average value of the measurements for each batch and whether that average value has significantly changed from batch-to-batch. Secondly, the R and S charts provide information about the short term variation found in the samples of each batch.

2.2.2 Control Charts for Profiles

Marsh and Tucker (1991) also suggest a way to apply control charts to batch profiles.

The principle idea is that the profile of a variable will maintain a general shape from batch to batch. This general shape can then be used as a target or standard profile.

The control charts used are based on the XBAR-R and XBAR-S charts that were discussed in the preceding section. One of the requirements of these control charts is that multiple readings of a variable must be made for each time unit of the profile. For example, a temperature measurement of a batch reactor may be taken every five minutes until the reaction is complete. But, each measurement consists of several readings or observations. Therefore, at each time unit of five minutes there is set of temperature readings. As can be imagined, the frequency of the temperature measurements may very well be limited by the amount of time required to take the

readings. The first step in creating these control charts is to establish a reference or standard profile. This standard profile is usually empirically determined by examining the shape of profiles from many past successful batches. The equations used by Marsh and Tucker (1991) are presented below with the “T” representing a temperature profile. First, the multiple readings for each batch are averaged at each time as follows:

$$T(t) = \frac{\sum T(n)}{n} \quad (1)$$

where: $T(t)$ is the average of the observations at time t

$T(n)$ is the observations at time t

n is the number of reading at each time t

The averaged measurements of each batch are then used by Marsh and Tucker to define a standard profile by averaging them together. This is represented by the following equation:

$$Ts(t) = \frac{\sum (T(t))(m)}{m} \quad (2)$$

where: $Ts(t)$ is the standard temperature at time t

$T(t)$ is the temperature at time t for batch m

m is the number of batches

After a standard profile has been established, then the differences between the standard profile and the profiles used to create the control charts are calculated by the following equation:

$$D(t) = T_m - T_s(t) \quad (3)$$

where: $D(t)$ is the difference at time t
 T_m is the profile measurement at time t
 $T_s(t)$ is the standard profile value at time t

Now, each time unit represents a subgroup of deviations from which an average and range or standard deviation can be calculated. It has been suggested by Marsh and Tucker (1991) that if the variation in the subgroups at each time remains constant, then the short term variation can be considered constant, and one set of control limits can be used for the entire temperature profile. If the variation is not constant, each time unit must be treated like an individual control chart with its own control limits. The assumption was made in this paper that the variation was constant and the control limits could be estimated from tables found in Duncan (1974). To use the control

chart, the standard profile values are subtracted from the individual readings for a batch at each time. The differences are then averaged and plotted on the chart.

When control charts are applied in this manner, they indicate two things. The Xbar chart shows how a profile deviates from a standard profile, which Marsh and Tucker describe as a long term effect. Secondly, the R and S charts indicate the short term variations found in a measurement at each time (t).

2.3 Multivariate Techniques Used to Monitor Batch Processes

Typically, there are many variables that can be used in the monitoring and control of batch processes. But, these variables are rarely independent of each other. They are usually correlated by the chemistry or physics of the process, and they may all be strongly time dependent. In order to gain a better understanding of the process and its dynamics, the relationships between variables must be understood.

Nokimos and MacGregor (1995) have developed a multivariate SPC approach that addresses the issue of correlated variables, and can be applied to batch processes. The method is based on the Multiway Principle Component Analysis of variable trajectories stored in process historians. As quoted by Nokimos and Macgregor (1994), the Multiway Principle Component Analysis (MPCA) is used to extract “the information in the multivariate trajectory data by projecting them onto low dimensional spaces

defined by latent variables or principle components”. To use the MPCA, the process data is setup as a three dimensional matrix. The dimensions are defined as time, batches, and measurements. This matrix is then decomposed into a summation of R products (principle components) of score vectors, loading matrices, and residual matrices. The MPCA decomposition is presented, by Nokimos and MacGregor, as the following equation:

$$\underline{X} = \sum_{r=1}^R t_r \otimes P_r + \underline{E} \quad (4)$$

where: $\underline{X}$ is the three dimension matrix of process data.

t_r is the score vectors which are related to the batches.

P_r is the loading vectors which are related to the variables and their time variance.

$\underline{E}$ is the residual matrices which are related to noise.

It is then stated that each element in the score vector (t) represents the overall variability of a batch compared to the variability of all other batches used in the analysis. The loading vector (P) is described, by Nokimos and MacGregor, as providing a summary of the “time variation of the measurement variables around their average trajectories, and its elements are the weights applied to each variable at each

time within the batch to give the t score for that batch". The information contained in the score and loading vectors can then be used too create multivariate SPC control charts.

Several advantages of using such a method have been identified by Nokimos and MacGregor (1994). First, all the information required to perform the analysis can be taken from the process' historical database. Next, the method maintains the concept of SPC by using past process data for the monitoring and evaluation of future process data. The information can also be presented in multivariate SPC charts that are interpreted in the same manner as conventional SPC charts, although the data plotted is not actual process data. Lastly, the multivariate approach provides more insight as to the cause of variation by identifying the variables and the contribution of those variables that are associated with deviations found in the multivariate control charts.

2.4 Method of Least Squares

The technique of analyzing profiles developed in this study uses a least squares analysis to solve an overdetermined set of equations. The least squares criterion allows solutions to be approximated when exact solutions do not exist. It also, as pointed out by Lawson (1974), can be used to provide "additional quantitative information describing the relationship of the solution parameters to the data". Least

square solutions can be found for both linear and nonlinear problems, both of which are discussed in the following sections.

2.4.1 Linear Least Squares

The basic linear least squares problem is discussed by Lawson (1974) and is presented as follows:

$$Ax = b \quad (5)$$

where: A is a $m \times n$ matrix with the rank $r \leq \min(m, n)$
 b is a vector of size m
 x is the least squares solution to the problem

The solution to the problem (x) is the one that satisfies the following equation:

$$\min_{(x_1, x_2, \dots, x_n)} \sum_{i=1}^m (A_i x - b_i)^2 \quad (6)$$

where: A is the set of equations to solve
 b is the respective solutions to the equations in A
 x is the least square solution
 m is the number of rows in matrix A and vector b

If A is overdetermined, the number of rows (m) will be greater than the number of columns (n). If the vector (b) is not a linear combination of the columns of (A), which means its not in the column space of (A), then there is no (x) that will produce the exact same vector (b). Therefore an (x) must be chosen such that it produces a vector that is as close to (b) as possible and still remains in the column space of (A). This introduces the concept of a projection which can be calculated as follows:

$$p = Ax = A(A^T A)^{-1} A^T b \quad (7)$$

where: P is the projection of the vector (b) onto the column space of matrix (A).

A is the column space to be projected onto.

b is the vector to be projected.

The least squares solution can be calculated when the columns of A are independent by the following equation:

$$x = (A^T A)^{-1} A^T b \quad (8)$$

where: x is the least square solution to the problem.

A is the column space to be projected onto.

b is the vector to be projected

2.4.2 Nonlinear Least Squares

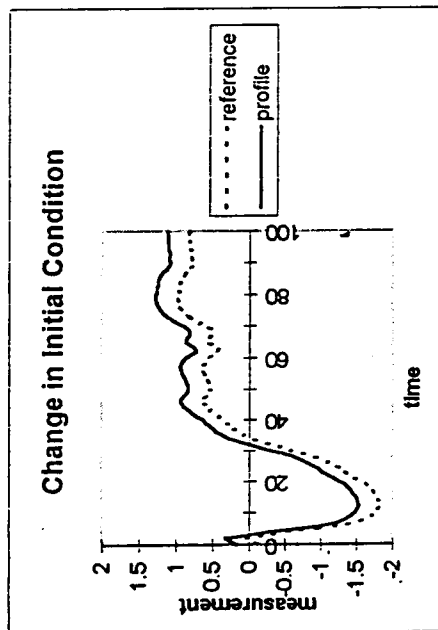
Nonlinear least square techniques were also considered in this study. A nonlinear least square problem is one that cannot be solved analytically. One method to solve nonlinear least square problems is to use a trial and error search routine for the least square solution (x) that minimizes a function that is based on the principle of least squares (minimization of error). An example of a search routine is the simplex method for minimizing a function presented by Nelder and Mead (1965). The method uses a relatively simple pattern in which to iterate the search parameters. It creates a simplex with $n+1$ points in n -dimensional space, where n is the number of variables in the function. With each iteration, it generates a new point to define a new simplex. The iterations continue until the minimum of the function is found. To increase the precision of the search, termination tolerances may be set on the function and the search variables. Note, this procedure was chosen because of its ease of use and availability in MATLAB. It has the disadvantage of finding a local minimum and may not be applicable to functions with more than one minimum.

Chapter 3

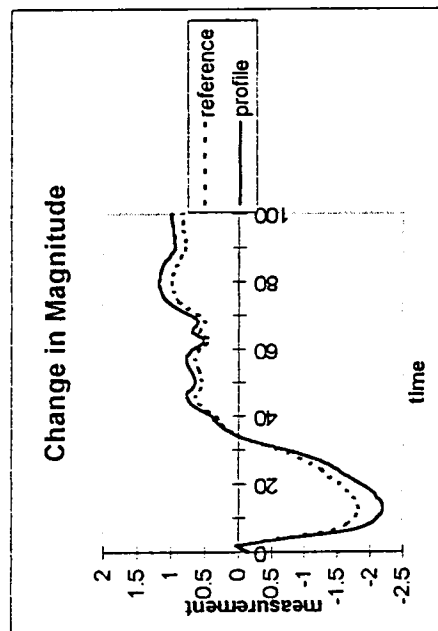
Least Square Profile Analysis

3.1 Summary of Technique

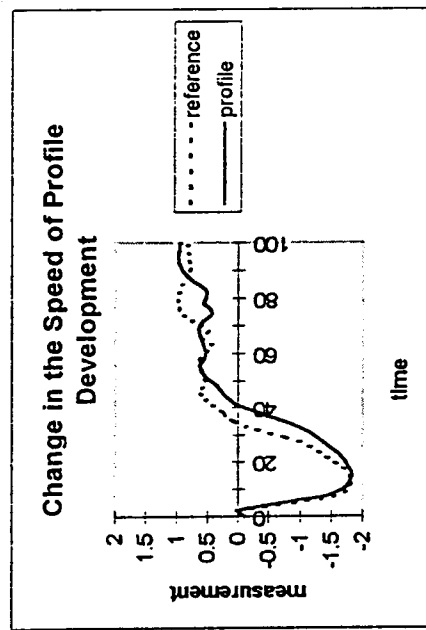
An SPC analysis of a batch process is quite useful in the identification of variation, but it alone provides little indication as to the cause of variation. This is where the application of least squares projections can greatly enhance an SPC analysis. The least squares projection step is used to preprocess profiles before applying any SPC techniques. The preprocessing step presented here provides a simple deterministic framework to quantify sources of variation. The principle assumption behind this technique is that the manifestation of variation in a profile can occur in several ways. In this study, three deterministic sources were considered and they are: a change in time scale (speed of profile development), a change in magnitude scale (the sensitivity of the profile), and a change in initial conditions. A fourth category also considered includes all "special cause" variability which cannot be explained by the three deterministic effects. Examples of how these forms of variations can alter a profile are presented in Figure 3.1.



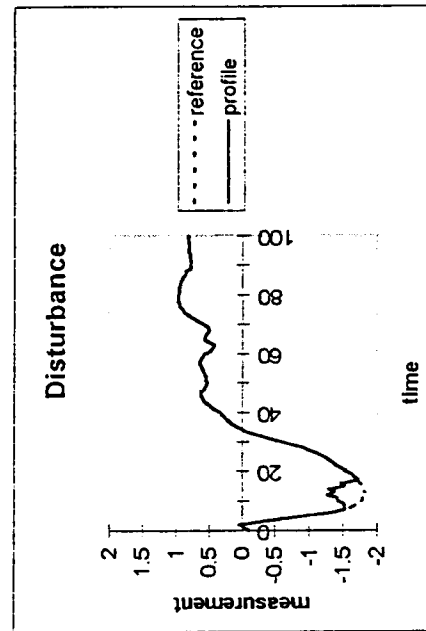
(a)



(b)



(c)



(d)

Figure 3.1 Effects of Variation on Profiles

3.2 Procedure for Analyzing Profiles

The procedure for analyzing profiles is relatively straightforward. A schematic of the procedure is presented in Figure 3.2. The first step is to collect the profiles to be studied. Next, a reference profile is projected onto the profile being studied, which creates a projection (adjusted profile) and a set of least square solutions (parameters). The projection and least square parameters are then plotted on control charts and analyzed. The information can then be stored in a historical database for further study. A detailed description of how the control charts are created is presented in the following sections

3.3 Procedure for Creating Control Charts

Several control charts have been used in this research. They are a profile control chart, a projection control chart, and a parameter control chart. Since both the profile and projection control charts are created in the same manner, the discussion will be limited to the projection and parameter control charts only.

3.3.1 Determining a Reference Profile

The determination of a reference profile is the first and most critical step in development of a projection control chart. The reference profile should contain the general shape and all characteristic features of the profiles produced by a batch. It is used as a basis from which all the projections are produced. The first step is to collect

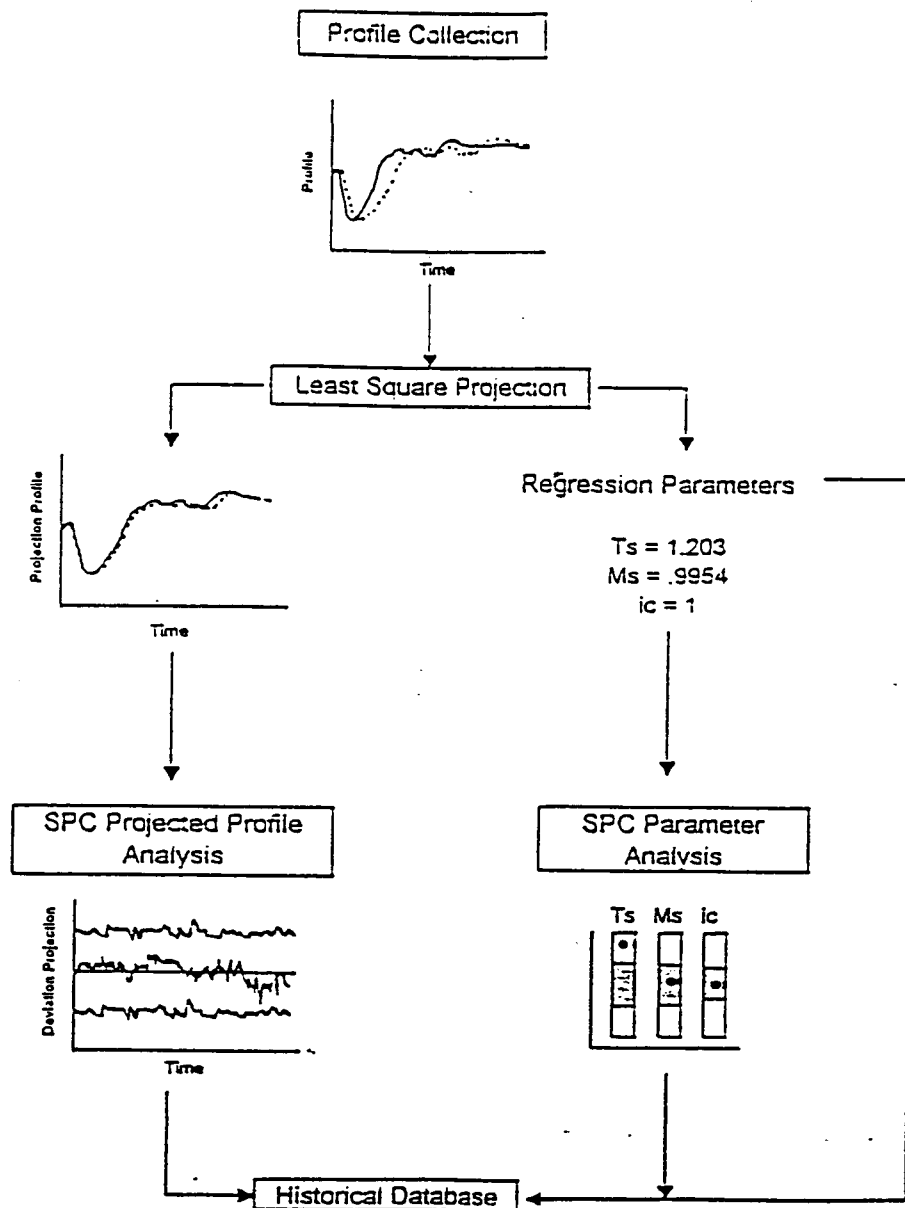


Figure 3.2. Flowsheet of the Procedure

a statistically acceptable number of profiles to be used as a basis for determining the reference profile. These profiles should be from batches that yielded an acceptable product and are consider to be typical in their operation. Once the profiles have been collected, the reference profile can be determined. This is accomplished by averaging the profiles, as discussed by Marsh and Tucker (1991), by the following equation :

$$Pr(t) = \frac{\sum(Pn(t))}{m} \quad (9)$$

where $Pr(t)$ is the reference profile value at time (t).
 $Pn(t)$ is the value of profile from batch (n) at time (t).
m is the number of profiles (batches)

This equation is used when there are single samples made at each time for each batch. If multiple samples are taken, an average profile must be calculated for each batch. Then equation (9) can be applied, where $Pn(t)$ is the average profile value at time (t) for batch (n). Once a reference profile has been established, the projection control charts can be constructed.

3.3.2 Projection Control Charts

The projection control charts are constructed by individually projecting the reference profile onto the profiles used in its determination. The projections are created by performing a nonlinear least square analysis where the least square solution is found by a trial and error search routine. The procedure for performing the nonlinear least square analysis can be broken into steps which are presented below.

1. *Extrapolation.* The first step is to extrapolate data for all the profiles that are used to create the control chart. The extrapolated data is used when the speed of the profile's development is slower than that of the reference profile. As the speed is adjusted during the least square analysis, the profile will be shifted in time creating an incomplete profile. The extrapolated points will be used as artificial data points and added to the end. An illustration of where the extrapolated points are added is presented in Figure 3.3. In the profile presented in this study, the reaction appears to be driven to a steady state, in which case the extrapolated data was taken to be the average of the last two points. The method of extrapolating data depends largely on the shape of the profiles being studied.

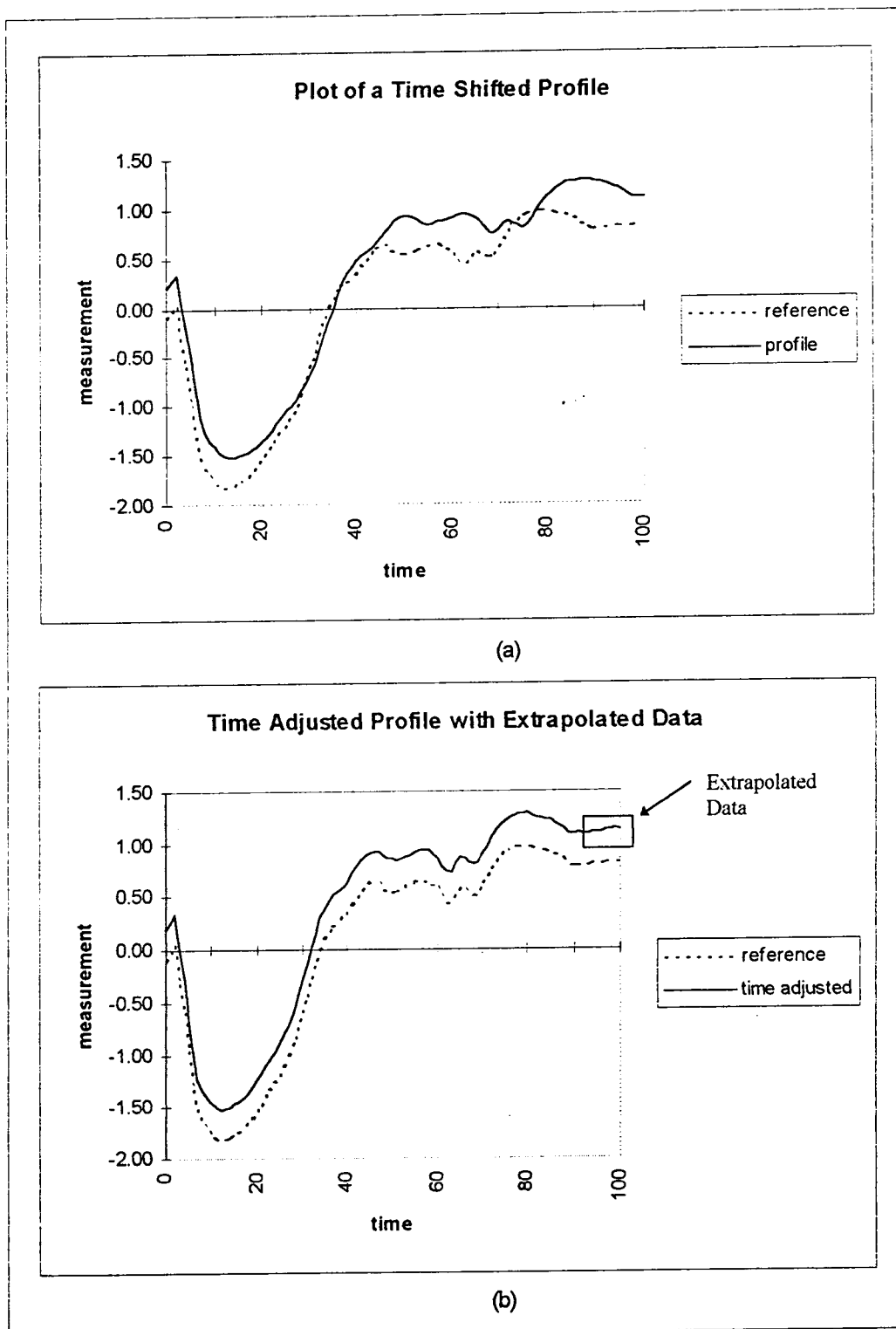


Figure 3.3 Example of the Use of Extrapolated Data

2. *Time Adjustment (Profile Speed)*. This is where the nonlinear least square analysis begins. The nonlinearity of this problem lies in the adjustment for time. It was found that if the profile was adjusted for time independently of the initial condition and magnitude the problem could be reduced to a linear least square analysis. To make this time adjustment, the time vector associated with the profile is multiplied by a time parameter. This shifts the profile in time. The problem can then be solved as a linear least square problem. A search routine is used to find the time parameter. The routine is based on a golden section search and parabolic interpolation. The time parameter is iterated on until the sum of the errors (or the distance between the projection and the reference profile) has been minimized. This particular search routine was chosen because of its availability and ease of use in MATLAB. A program using the same algorithms can be found in Forsythe (1976).

3. *Interpolation*. After the time vector has been multiplied by a time parameter (or the profile shifted in time), the profile must be interpolated. In this analysis, the reference profile and the profile of interest must share the same time vector. So, if there is a measurement for the reference at time 10, there must be a measurement for the profile at time 10. Therefore, the profile values are interpolated for the reference profile time vector. This step also includes the truncation of the time adjusted profile,

if necessary. The profiles being analyzed must have the same number of measurements (or elements in their vector) as the reference profile.

4. *Linear Least Squares Analysis.* The next step is to perform a linear least square analysis on the time adjusted profile to create a projection and to solve for the initial condition and magnitude parameters. The parameters are determined by the following equation:

$$x = (A^T A)^{-1} A^T b \quad (8)$$

where: A is a matrix consisting of a column of ones and the time adjusted
 profile values.

 b is a vector of the reference profile values

 x is a vector containing the magnitude and initial condition parameters

Once the remaining parameters have been determined, a projection can be produced by the following equation:

$$p = Ax \quad (7)$$

where: p is the projection vector

A is a matrix consisting of a column of ones and the time adjusted
profile values

x is a vector containing the magnitude and initial condition parameters

5. *Iteration.* After a projection vector has been determined, the sum of the errors between the projection and the reference profile is calculated. A new time parameter is chosen until the sum of the errors has been minimized (least square solution has been found).

After the reference profile has been projected onto the profiles used in its determination, there should be a projection and a set of least square solution consisting of three numbers for each profile. Based on these projections and least square solutions the control limits for the control charts can be calculated.

Note. A multivariable search, as described in Section 2.4.2, was also used to create the projections, but the method described above was far more time efficient and produced the same results.

3.3.3 Calculating Control Limits for the Projection Control Chart

The control limits were created by treating each time unit as an individuals control chart. Therefore, each time will have its own individual control limits. The standard

deviation of the projections values was calculated for each time unit by the following equation:

$$s = \sqrt{\frac{\sum_i^n (P_i - \bar{P})^2}{n - 1}} \quad (10)$$

where: $\bar{P}$ is the average of the projection values at time t

P_i is the individual projection values

n is the number of projections

The use of equation (10) contradicts much of the literature found on individual control charts. Typically, the standard deviation for individual control charts is calculated using the average moving range as shown in the following equation:

$$\sigma = \frac{\overline{MR}}{d_2} \quad (11)$$

where: $\overline{MR}$ is the average moving range

$d_2 = 1.128$ for a subgroup size of 2

The limits calculated by equation (11) are not applicable in this study because the profiles used to determine the limits were not time ordered. Once the standard

deviation (s) were calculated, the upper and lower control limits for each time unit were defined as $\pm 3s$. Two control charts are presented in Figure 3.4. Both charts present the same information, except the reference profile has been subtracted from the projections in Figure 3.4 (b). In doing so, the time variant nature of the projections can be taken into account and the information can be neatly presented as deviations from the reference profile. It may be noted that these control charts are not the same as the difference control charts discussed by Marsh and Tucker (1991).

3.3.4 Parameter Control Charts

Just as the projections can be used to analyze profiles, the least square solutions of the projections (adjustment parameters) can also be used in a profile analysis. Every projection used to determine the control limits for the projection control chart is accompanied by a set of three parameters (the least square solutions). This collection of parameters can be used to develop parameter control charts. The standard deviation of the parameters were calculated, respectively, by equation (10), and the upper and lower control limits were defined as $\pm 3s$. The parameters can then be plotted on charts like the ones presented in Figure 3.5. The “Ts” represents the time (speed) parameter, the “Ic” represents the initial condition parameter, and the “Ms.” represents the magnitude parameter.

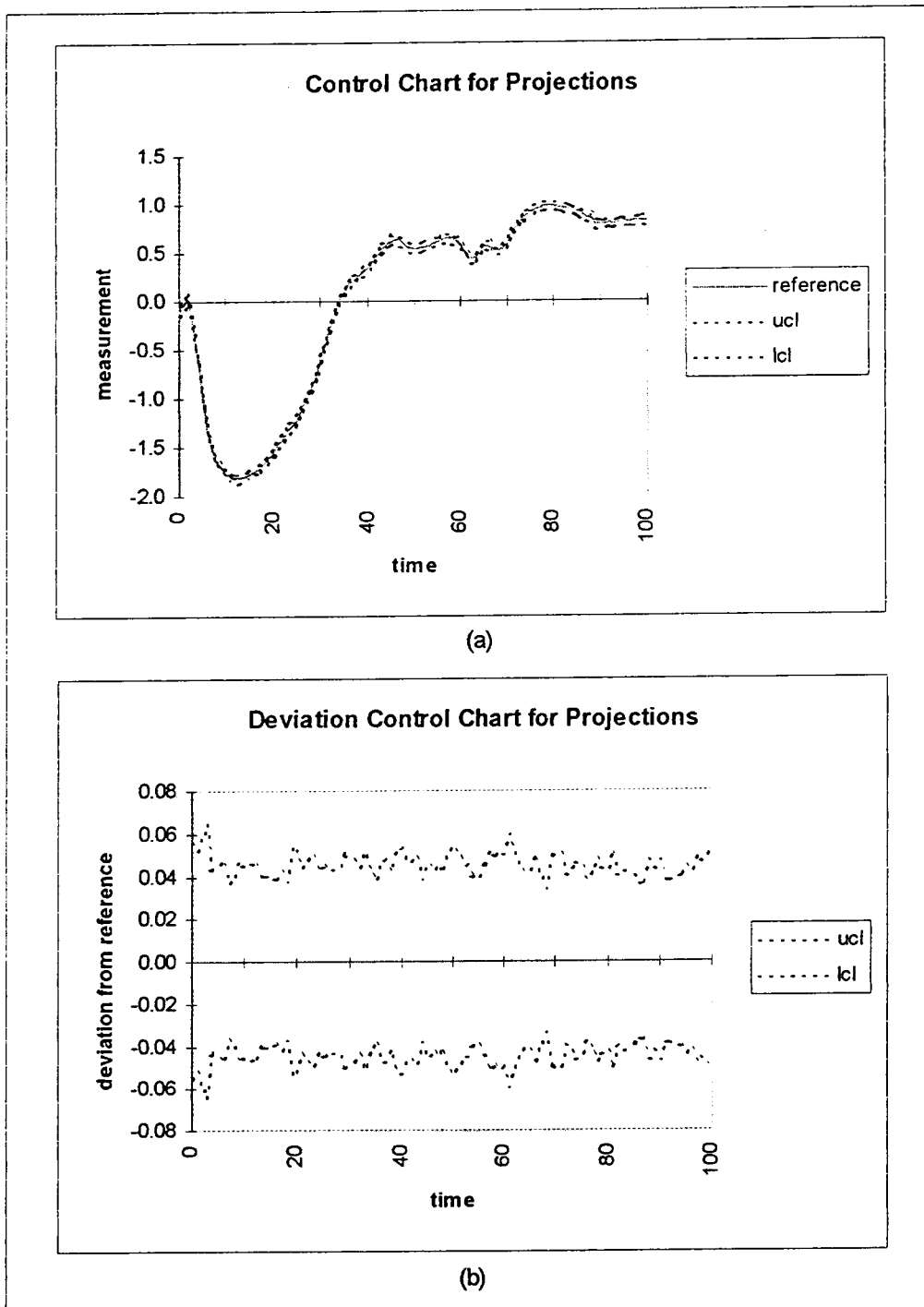


Figure 3.4 Two Types of Control Charts for Projections

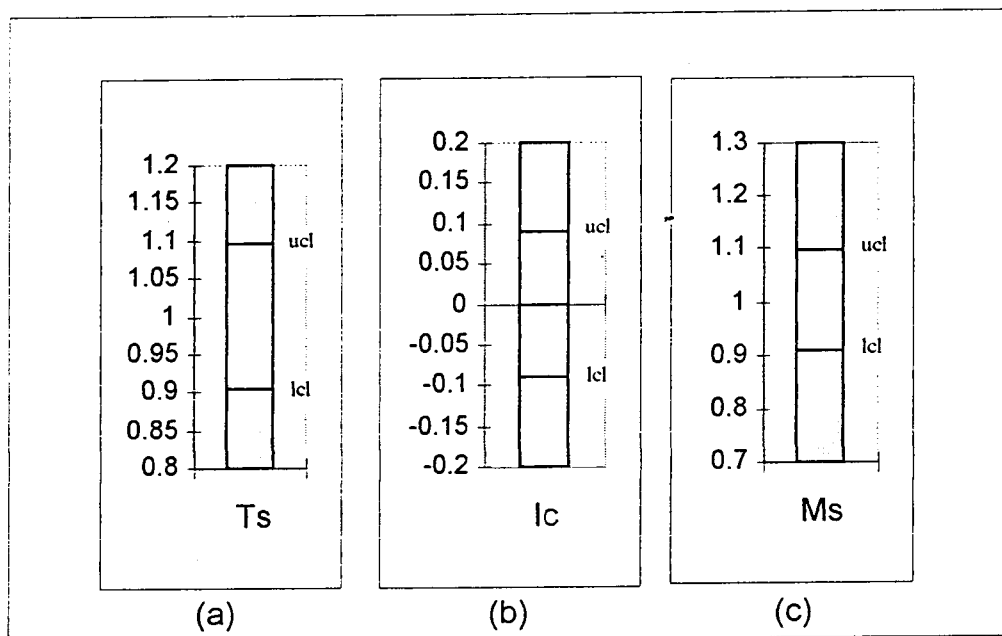


Figure 3.5 Parameter Control Chart

3.4 Dealing with Time During a Least Square Analysis

In developing this technique, the adjustment for time or the speed of profile development was not as straightforward as it first appeared. There are several ways the matrix A in equation (8) can be set up to adjust for time. The first way considered was to define the matrix in two columns; one column is a series of ones that represents an intercept, while the other column is the profile values. The column of ones is used to correct for changes in the initial conditions, and the column of profile values is used to correct for a change in magnitude and the speed of profile development. To test this setup, the reference profile was projected onto a profile that had a change in initial condition, an increase in magnitude, and an increase in the speed of the profile development. Figure 3.6 shows both the profiles and the resulting projection. In this idealized example, all the disturbances should be removed by the projection step if the method works correctly. Therefore, the projection should fall directly on the reference profile. As can be seen in Figure 3.6 (b), the projection does not fit the reference profile very well. This is because the least squares was unable to adjust for the profile development speed.

Another approach is to add the time vector to the profile matrix (A). The time vector is made up of the times each measurement was taken from the beginning of the batch. Now, the matrix (A) contains three columns; the column of ones, the time, and the

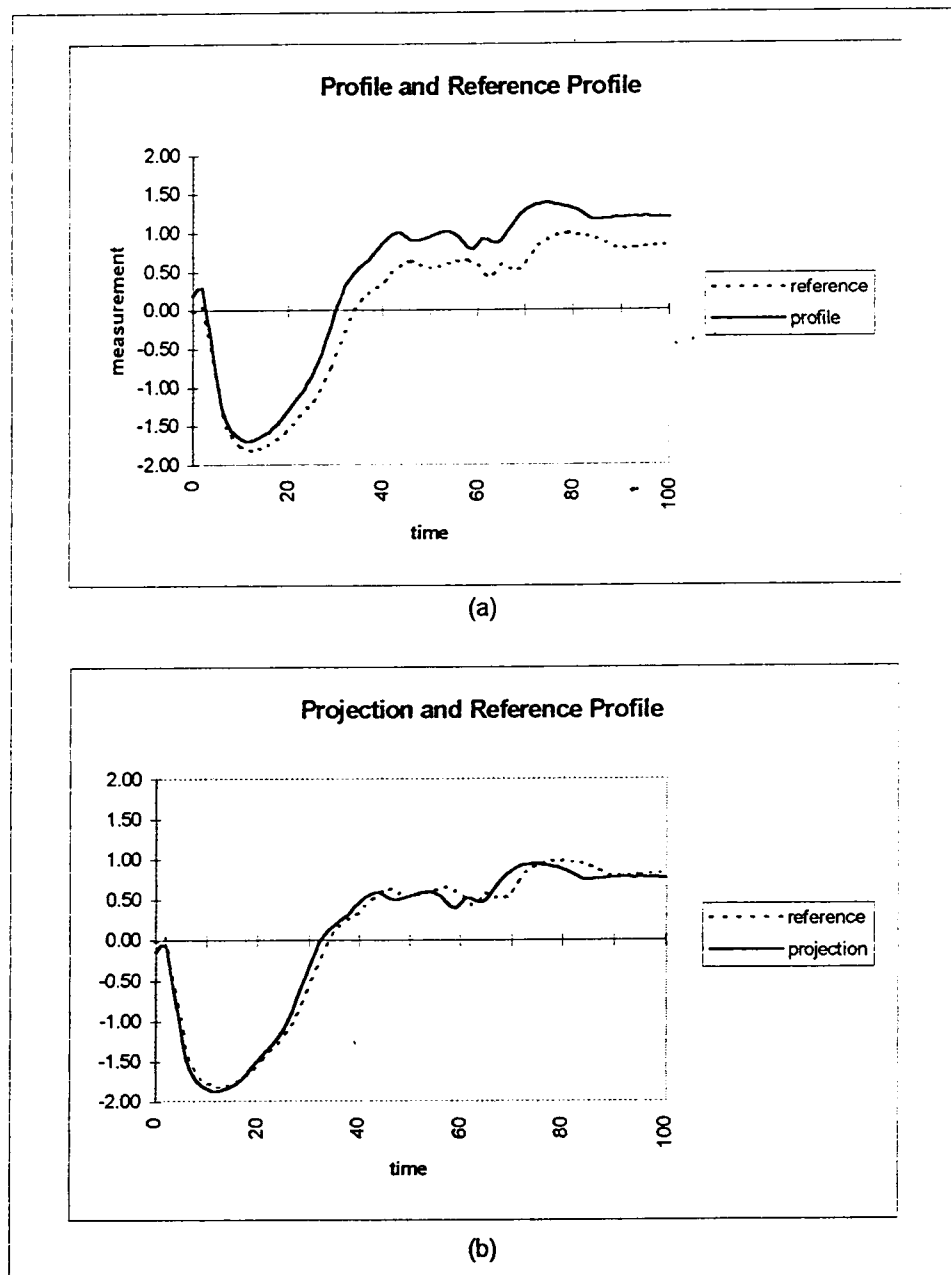


Figure 3.6 Unsuccessful Projection without a Time (Speed) Adjustment

profile values. The results are presented in Figure 3.7. Once again the projection step fails to adjust properly for the speed of profile development.

Based on the findings above, it was determined that the speed of profile development should be adjusted independently from the magnitude and initial conditions. The profile is first adjusted for speed and then the least square technique is applied for the magnitude and initial condition adjustments. The time adjustment was made by multiplying the profile's time vector by the time adjustment parameter. Then, the resulting time adjusted profile was interpolated to obtain profile values for the reference time vector. The details of this procedure are outlined in Section 3.3.2, and the results are presented in Figure 3.8. As can be seen in Figure 3.8 (b), the projection falls directly onto the reference profile. Although this method is not as straightforward as a simple matrix operation, it retains the fundamental idea of least squares and produces the same results.

3.5 Comments on Determining a Reference Profile

The method of calculating the average of a collection of "good" profiles is typically an adequate way of determining a reference profile, but there are certain circumstances when an alternative method may be more suitable. An example would be when profiles have distinct characteristic features that may be altered, or lost, in the averaging of the profiles. Figure 3.9 shows a collection of profiles that are used to

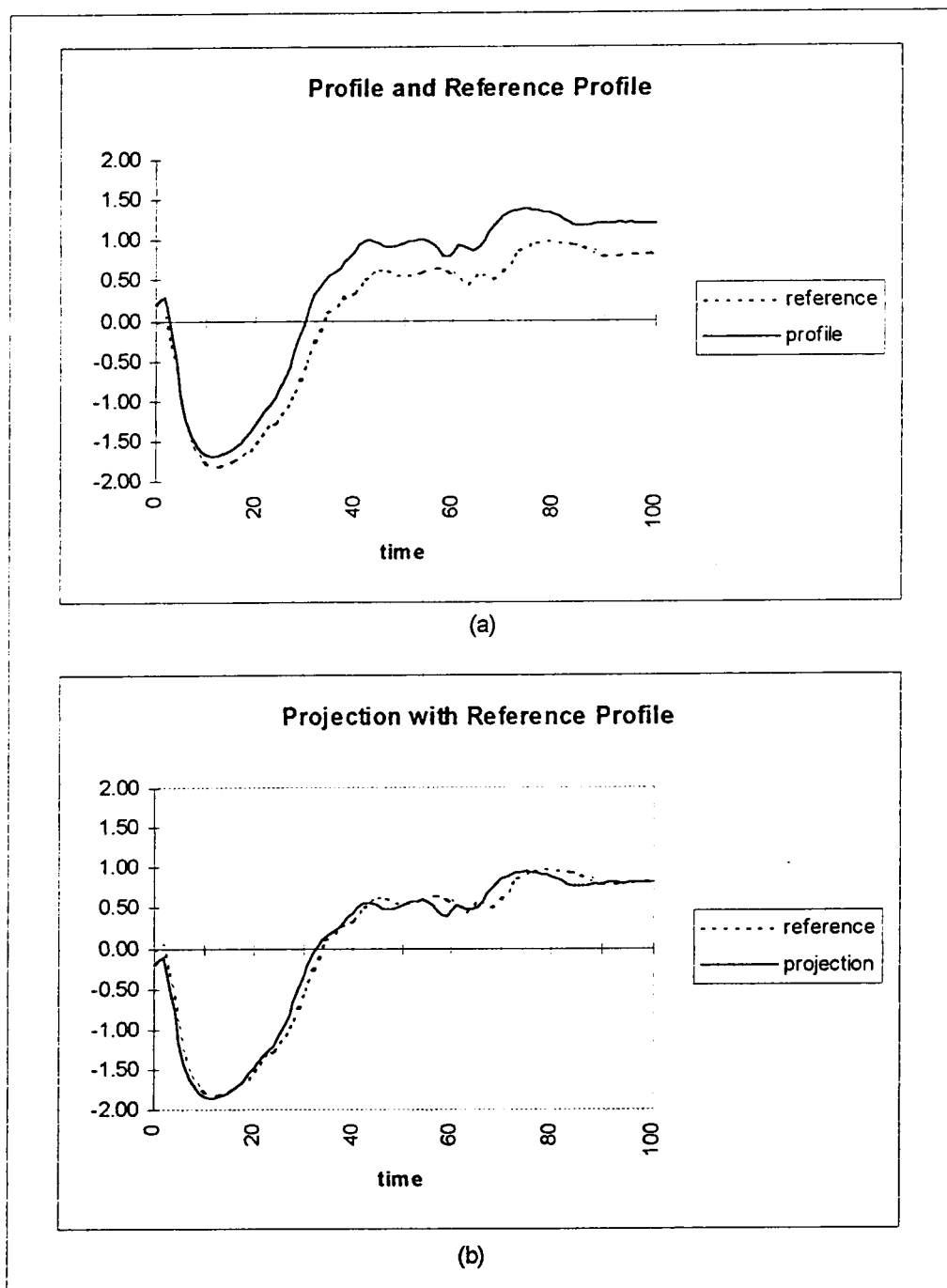
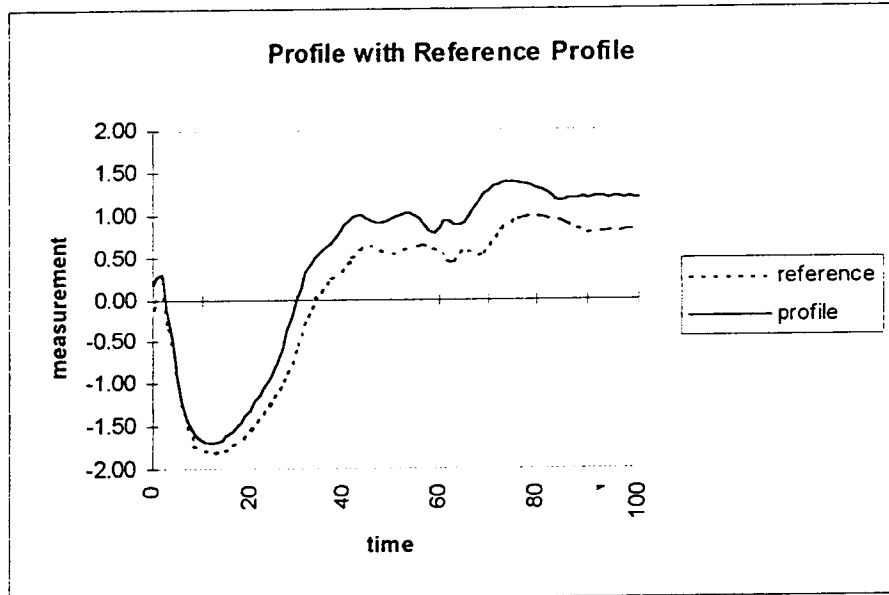
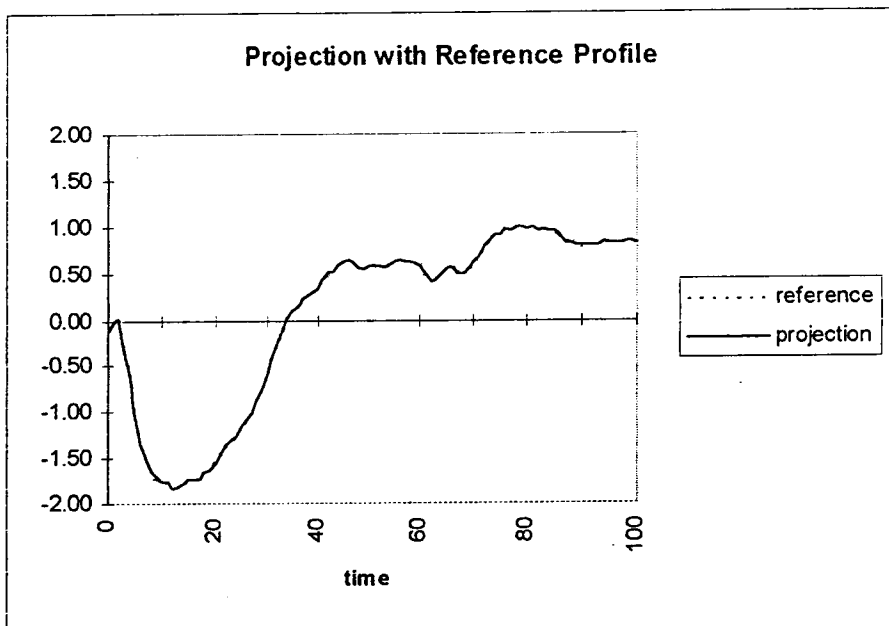


Figure 3.7 Unsuccessful Projection with a Time (Speed) Adjustment



(a)



(b)

Figure 3.8 Successful Projection with a Time (Speed) Adjustment

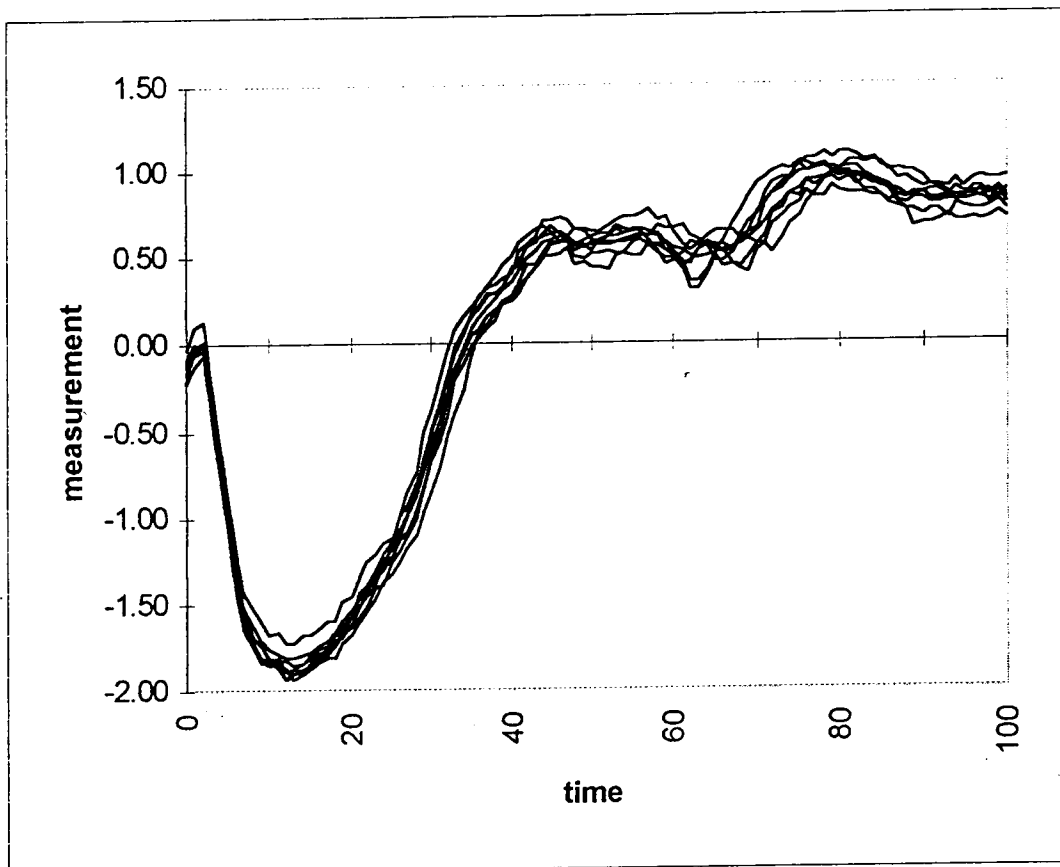
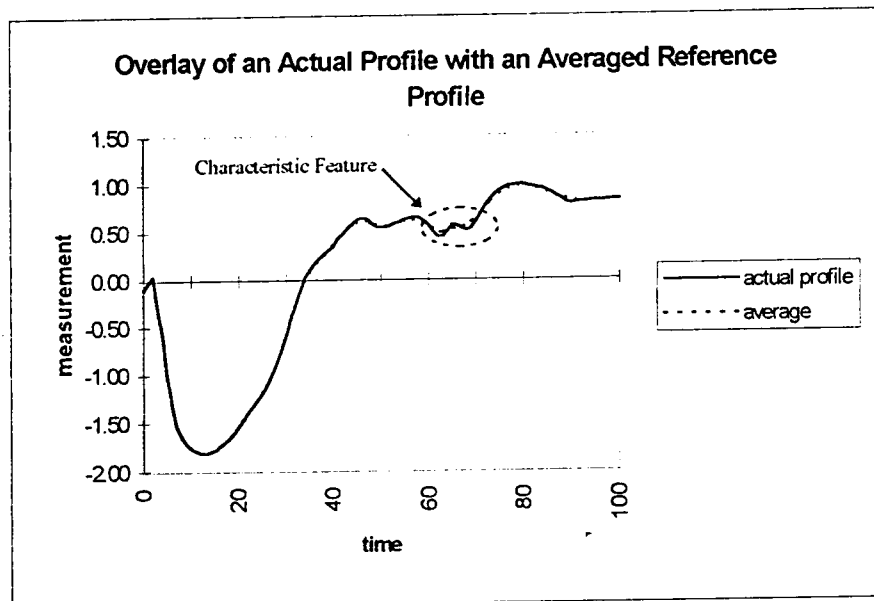


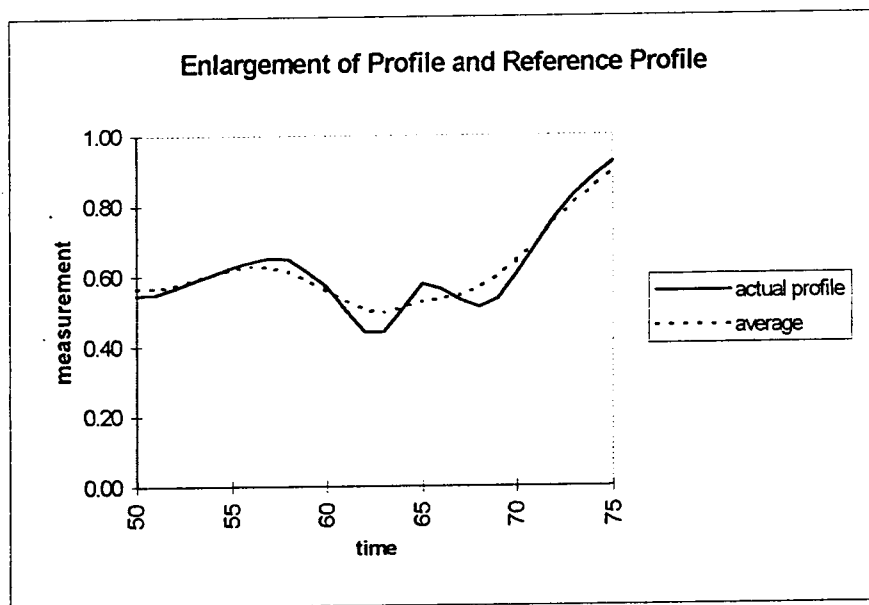
Figure 3.9 Collection of Profiles Used to Determine a Reference Profile

calculate a reference profile by the averaging method. The profiles contained in this collection all share the same general shape. The averaged reference profile and a profile from the collection are plotted in Figure 3.10. In Figure 3.10 (a), it can be seen that the reference profile does indeed describe the general shape of the profiles, but it fails to capture a characteristic feature found in all the profiles created by the batch. An enlargement of this feature is shown in Figure 3.10 (b). As can be seen, the reference profile has lost the feature and therefore may not adequately represent the profiles produced by the batch. A misrepresentative reference profile could lead to “false” control chart violations. These violations would most likely appear in the projection control charts because of the errors produced in the least square analysis of the profiles.

An alternative method for determining a reference profile might be to calculate an average profile and then choose a actual profile that most closely resembles the average. This is not a very elegant approach, but it may suffice for the lack of a better method.



(a)



(b)

Figure 3.10 An Actual Profile with an Averaged Reference Profile

Chapter 4

Examples of a Profile Analysis

In an effort to evaluate the usefulness of this technique, a collection of profiles were created based on a batch profile found in the literature. The profile used came from Nokimos and MacGregor (1995) and is presented in Figure 4.1.

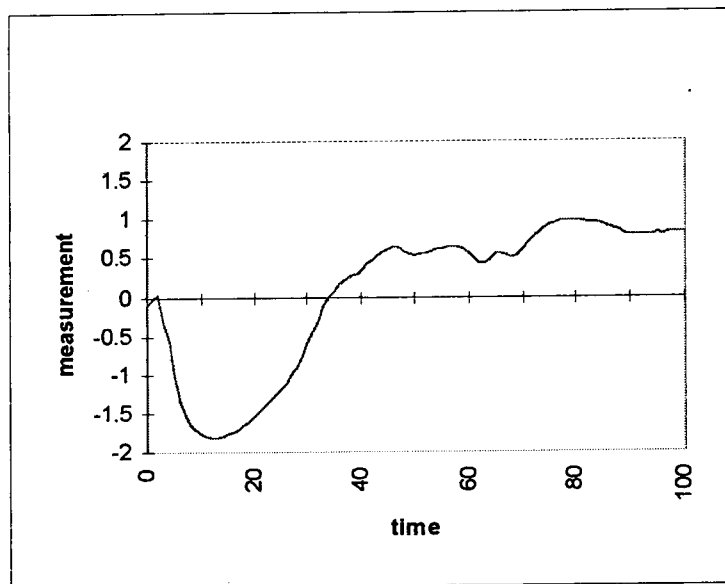


Figure 4.1 Profile Used for the Analysis

From this one profile, which was used as the reference, a collection of thirty profiles were produced. Each profile was altered from the reference profile by changes in: initial condition, magnitude, speed of profile development, and random noise. The

profiles were generated to give approximately the same profile variability as that found in the paper by Nokimos and MacGregor (1995). Based on this collection, both a conventional profile and projection SPC control chart were created as described in the previous chapter.

Once the control charts were created, several additional profiles were constructed to test the technique's effectiveness. These profiles contained forms of variation that the technique should and should not be able to quantify and explain. The analysis of four such profiles is presented in this chapter. In each case, a convention profile control chart will be contrasted with a projection control chart.

4.1 Analysis of Profile 1

The first profile is presented in Figure 4.2. By visual inspection, the profile seems to develop more quickly than the reference profile, and there appears to be no other abnormalities. After analyzing the profile, it becomes clear as to what form of variation is present and whether it is a non-typical batch. In Figure 4.3, both the profile deviation control chart and the projection deviation control chart are presented. The profile control chart in Figure 4.3 (a) identifies the batch as containing a non-typical amount of variation by violating the control limits. However, the control limits on the projection control chart in Figure 4.3 (b) are not violated. This simply means that the variation present can be quantified and explained by the preprocessing step. The parameter control charts are presented in Figure 4.4. As expected, the time

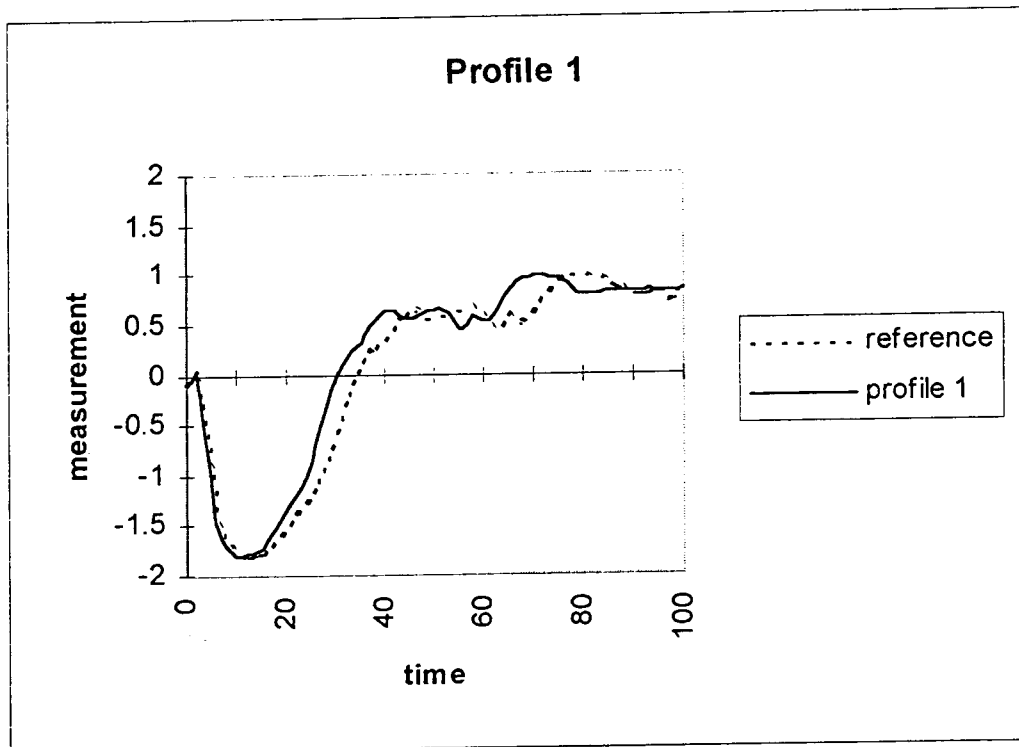


Figure 4.2 Plot of Profile 1 with the Reference Profile

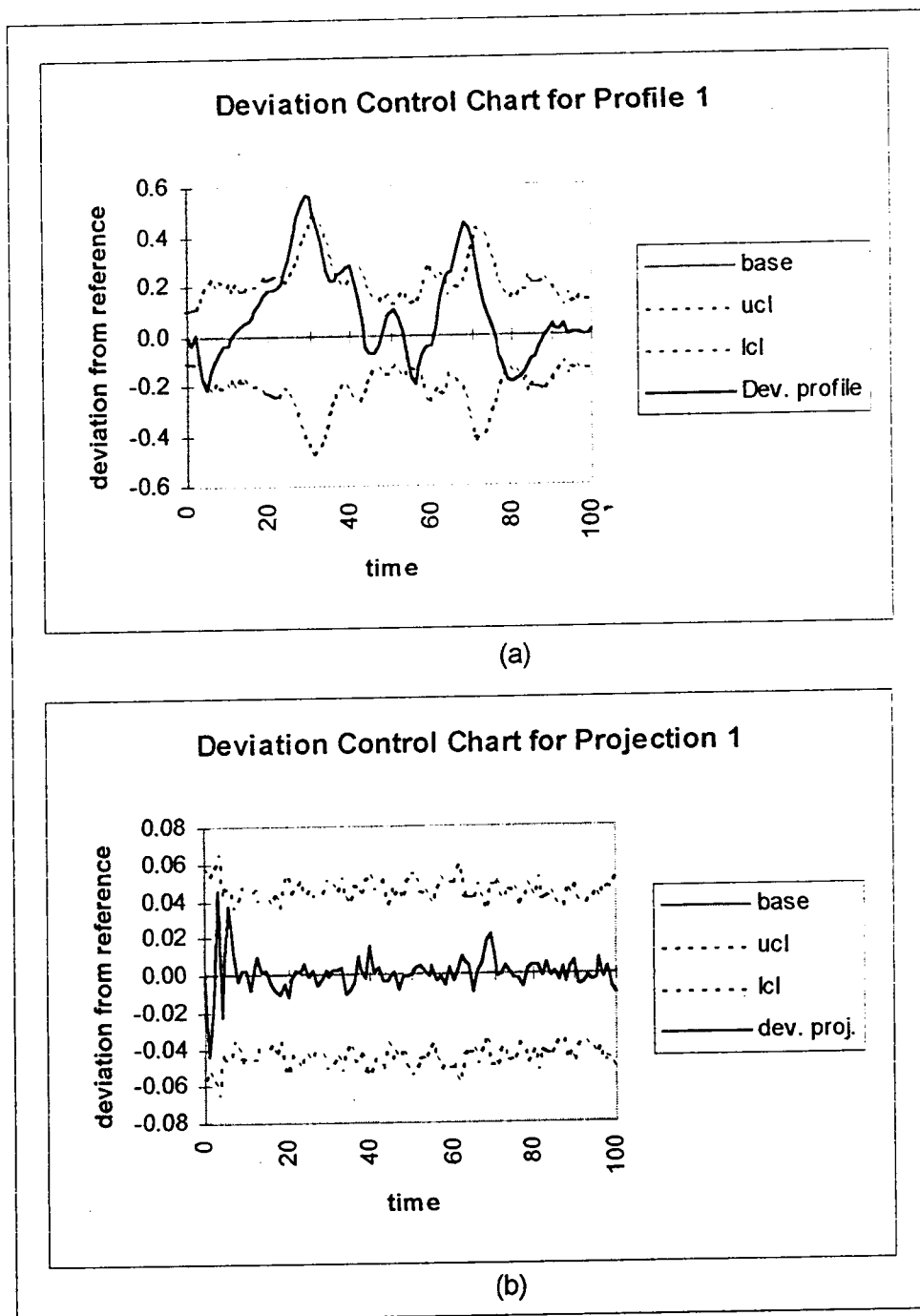


Figure 4.3 Control Charts for Profile 1.

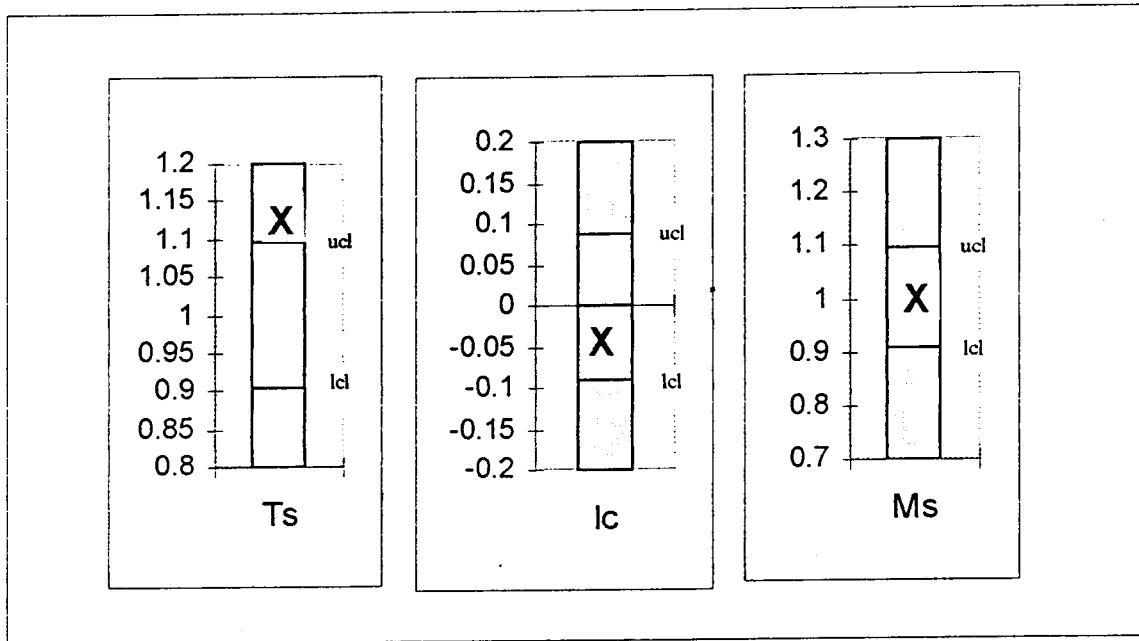


Figure 4.4 Parameter Control Chart for Profile 1

(speed) parameter (T_s) violates its control limits while the remaining parameters remain in control. A violation of the parameters control limits without a violation of the projection control chart signifies that the variation can be explained as one of the three manifestations and that the batch is non-typical in operation.

4.2 Analysis of Profile 2

The second profile analyzed is presented in Figure 4.5. The profile appears to contain a change in magnitude relative to the reference profile. The control charts are presented in Figure 4.6. The profile control chart and the projection control chart show no violations of their control limits. But, the control limits for the magnitude parameter control chart (M_s) in Figure 4.7 is violated. This means that the profile can be characterized by the preprocessing step and it contained an unusual amount of variation in its magnitude.

4.3 Analysis of Profile 3

The analysis of Profile 3 proved to be more interesting than the previous two. The profile is presented in Figure 4.8. The profile follows the reference profile but deviates at several points along the way. The control charts are presented in Figure 4.9. The conventional profile control chart in (a) shows no violations of the control limits which could imply that no unordinary variation exists. The projection control chart in (b), however, is violated at multiple points. This means that the preprocessing step was unable to characterize all of the variation contained in the profile. Therefore, the

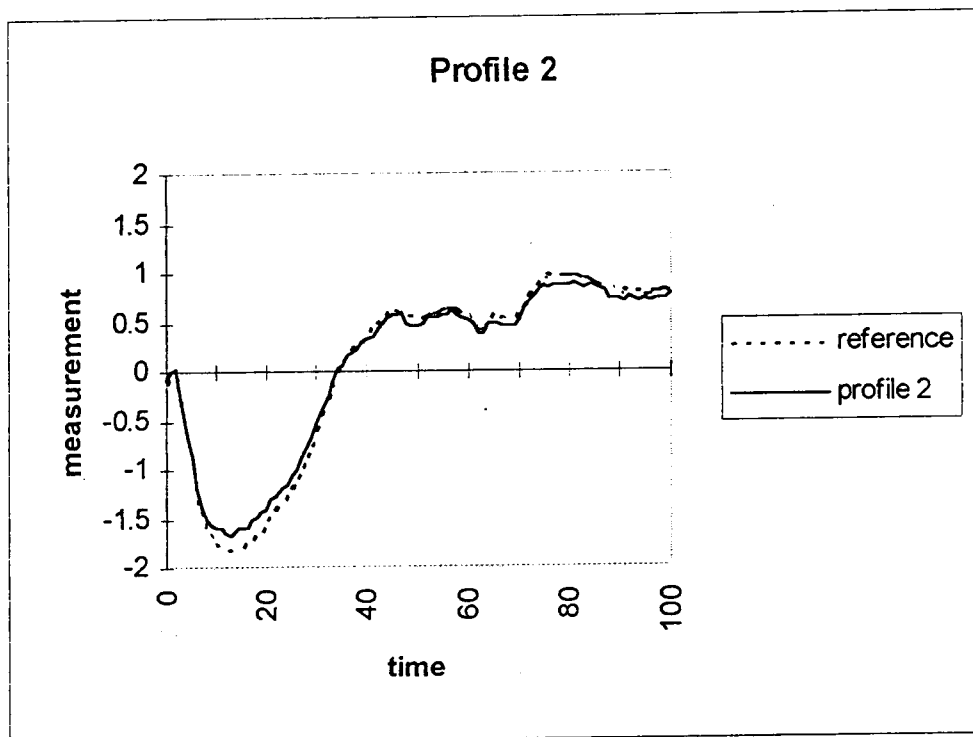


Figure 4.5 Plot of Profile 2 with the Reference Profile

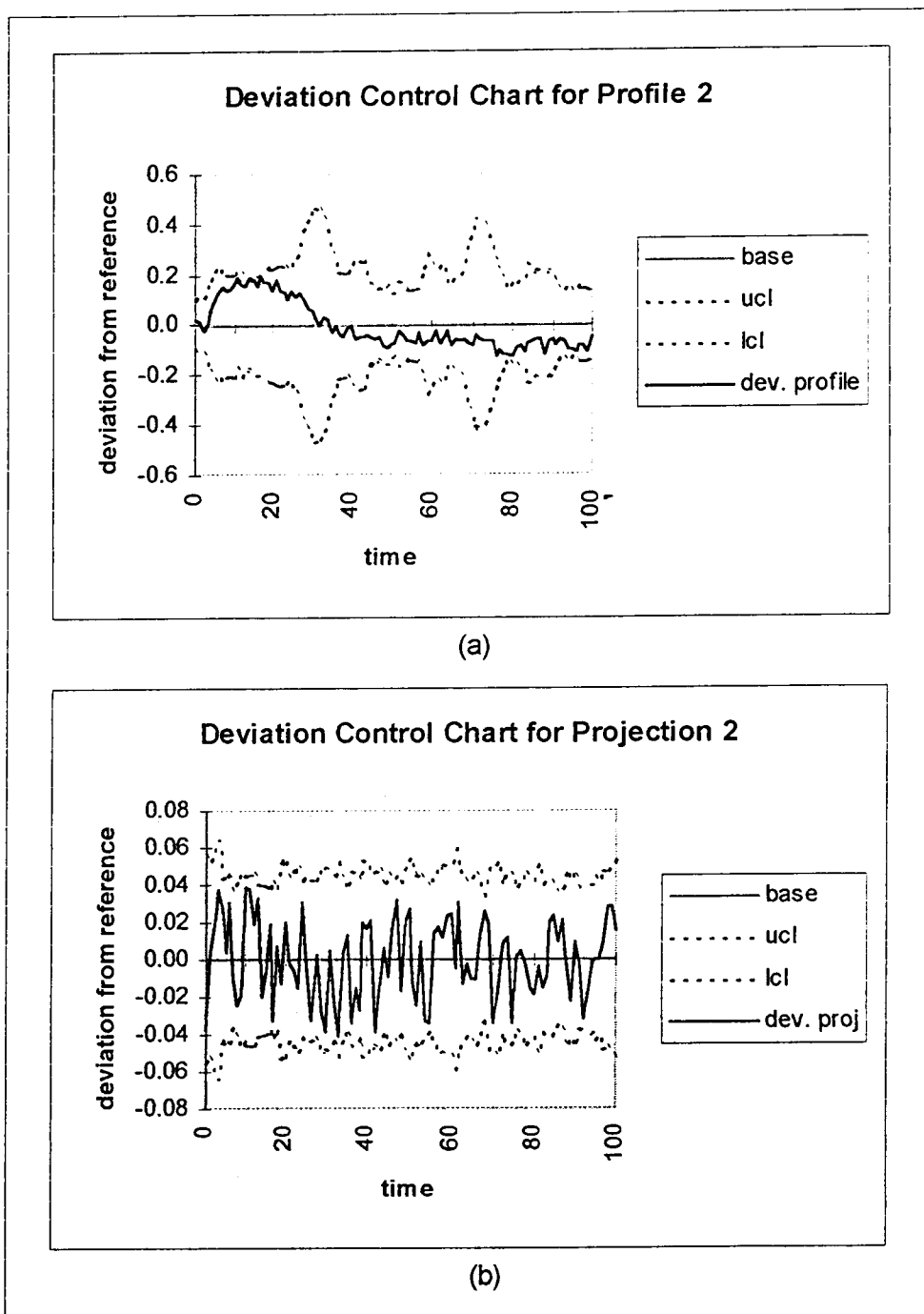


Figure 4.6 Control Charts for Profile 2

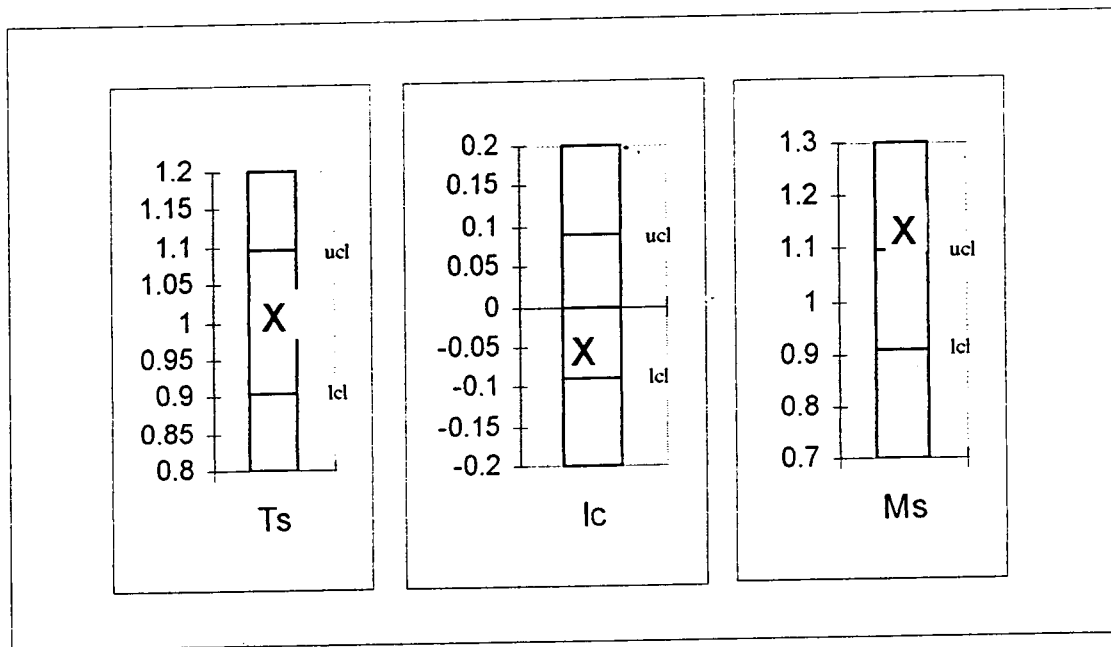


Figure 4.7 Parameter Control Chart for Profile 2

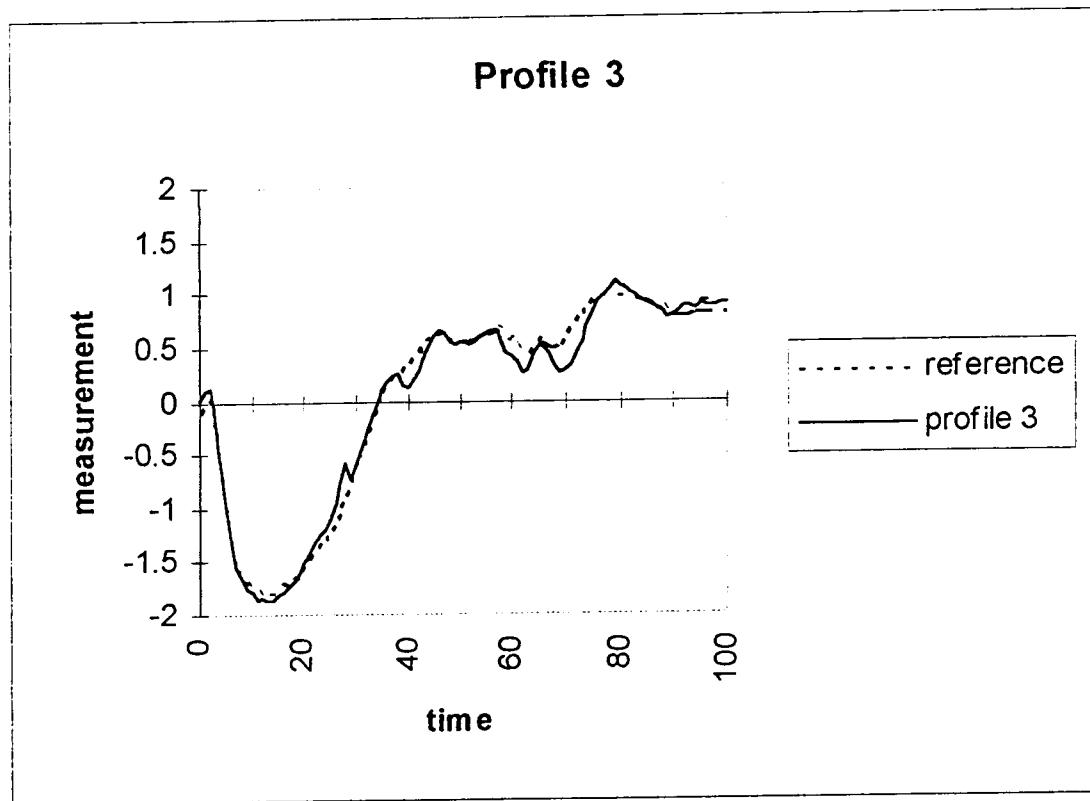


Figure 4.8 Plot of Profile 3 with the Reference Profile

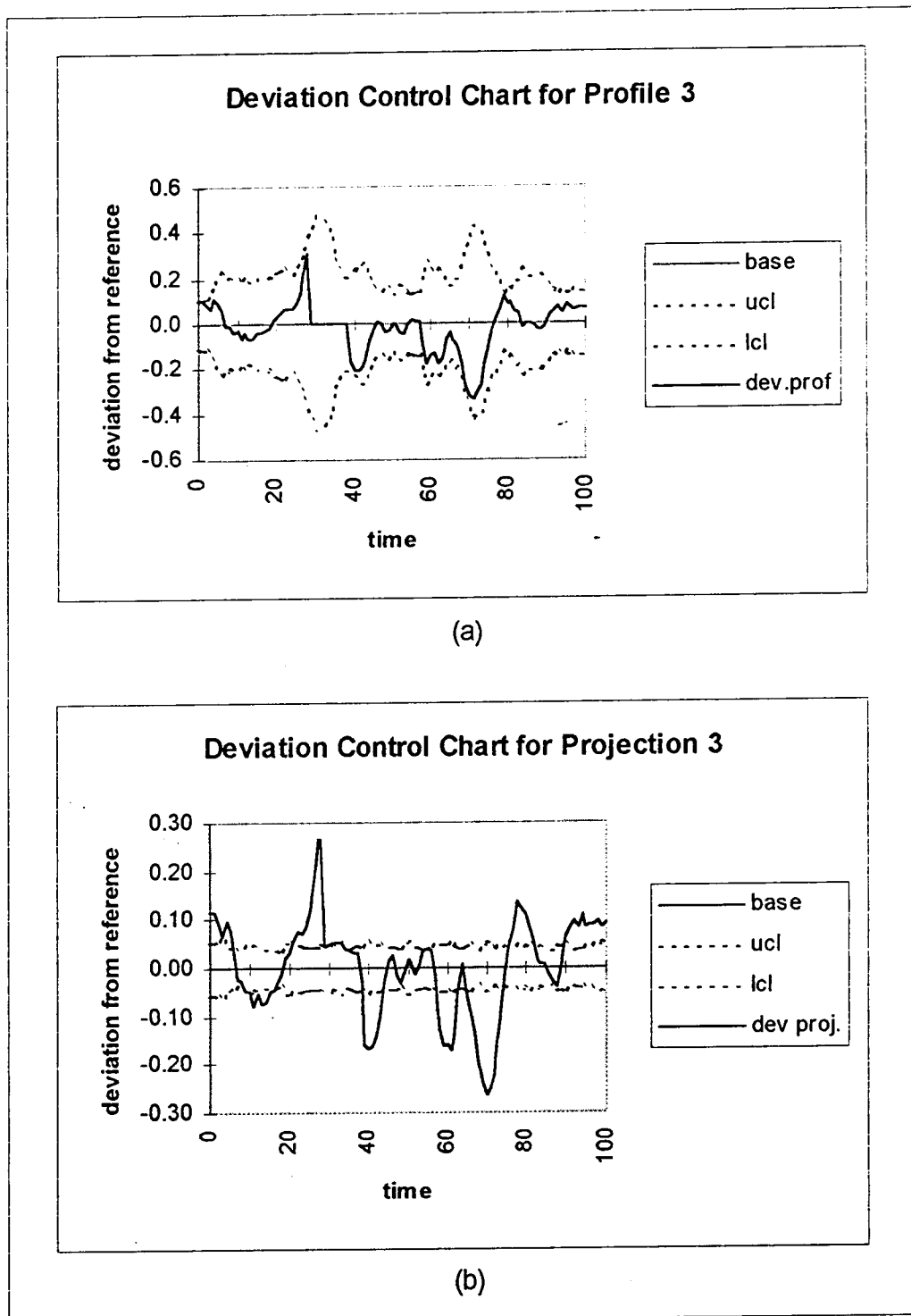


Figure 4.9 Control Charts for Profile 3

variation can be concluded to be caused by some "special cause". It may be noted that the parameter control charts in Figure 4.10 are not violated. This simply means that the variation in initial condition, magnitude, and speed of the profile was not out of the ordinary.

4.4 Analysis of Profile 4

The final profile is presented in Figure 4.11. The profile appears to have the same general shape as the reference profile, but seems to be shifted down at several points. The control charts for this profile are presented in Figure 4.12. Both the profile and the projection control charts are violated. This means that a significant amount of variation exists in the profile that is not characterized by the preprocessing step. The parameter control charts are presented in Figure 4.13. Of the three parameters, the control limits for the initial condition parameter (I_c) found in (b) are violated. This violation along with the violation of the projection control chart implies that a significant amount of variation did manifest itself as a change in initial conditions, and that the profile also contains "special cause" variability.

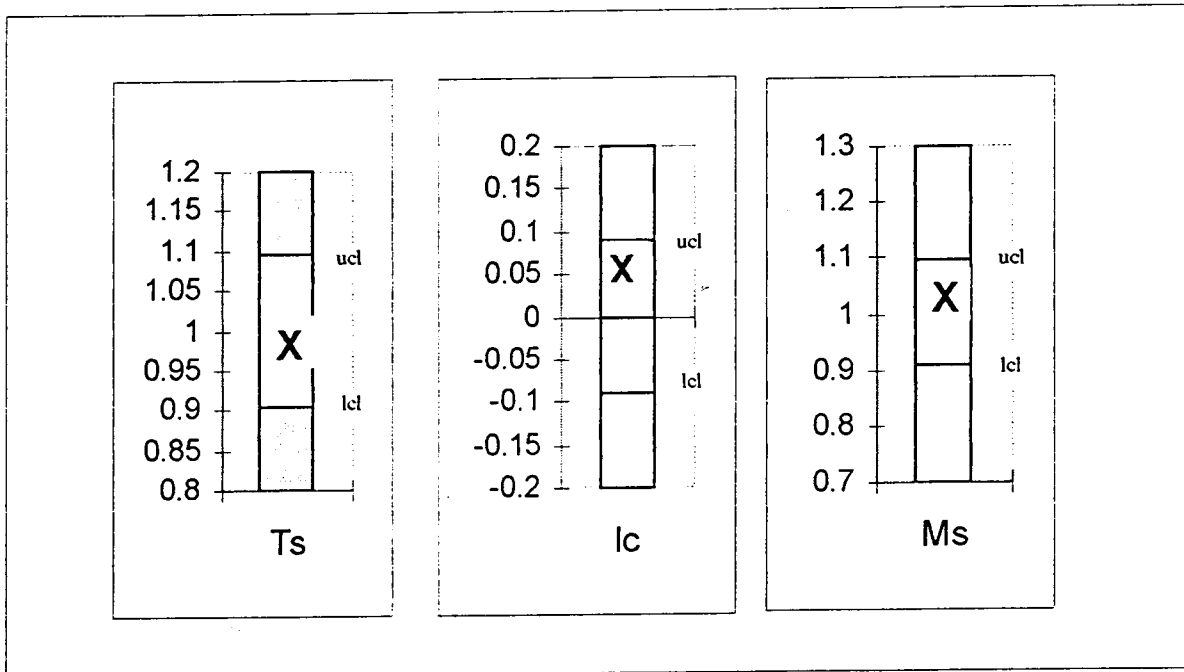


Figure 4.10 Parameter Control Chart for Profile 3

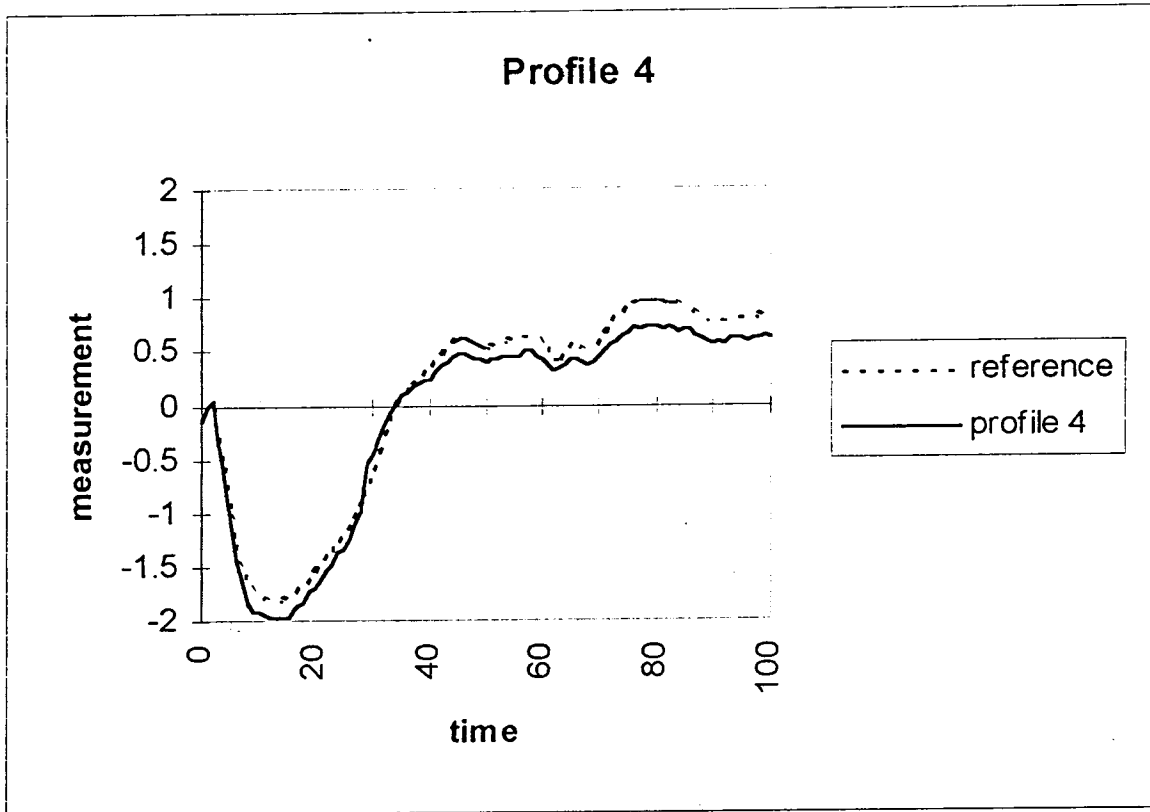


Figure 4.11 Plot of Profile 4 with the Reference Profile

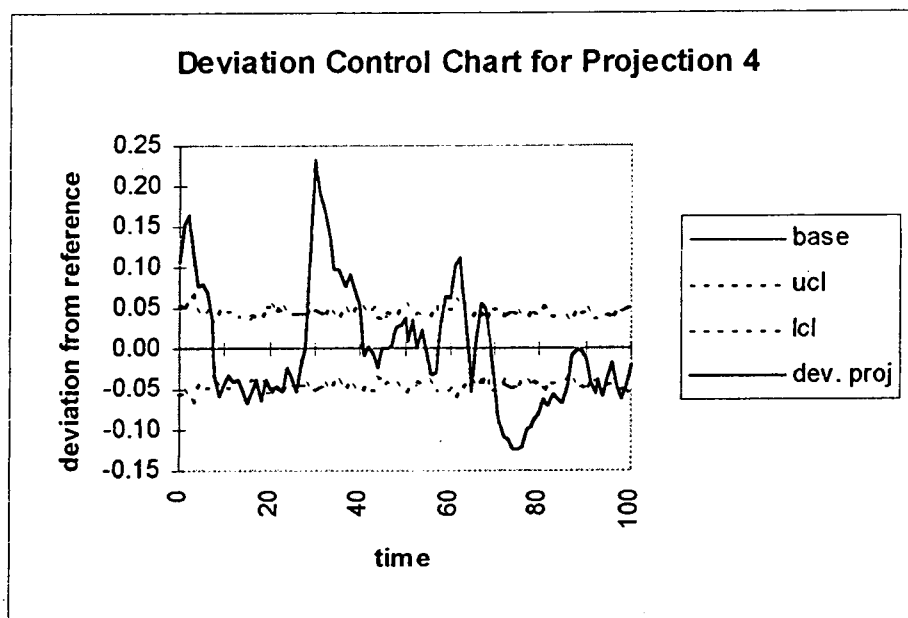
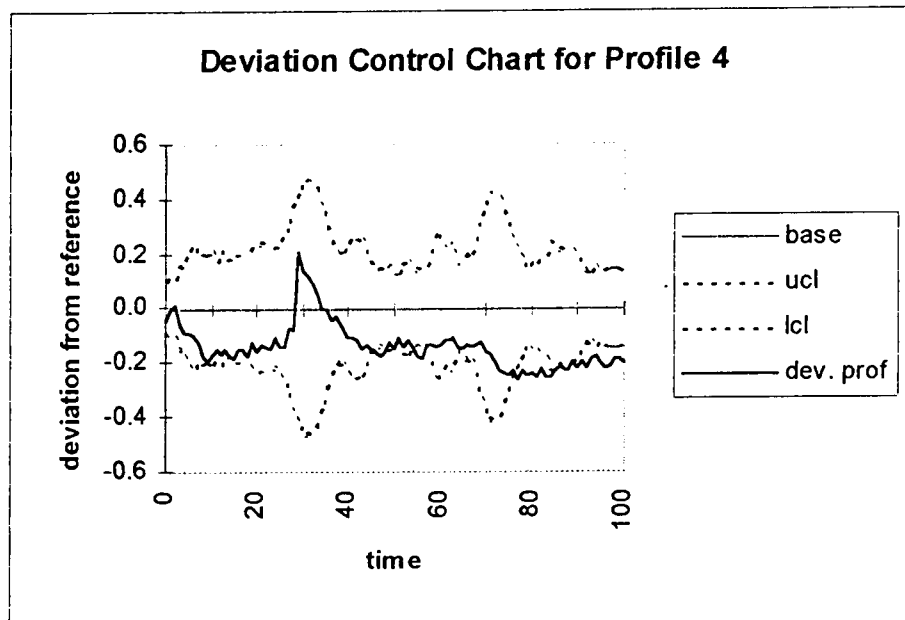


Figure 4.12 Control Charts for Profile 4

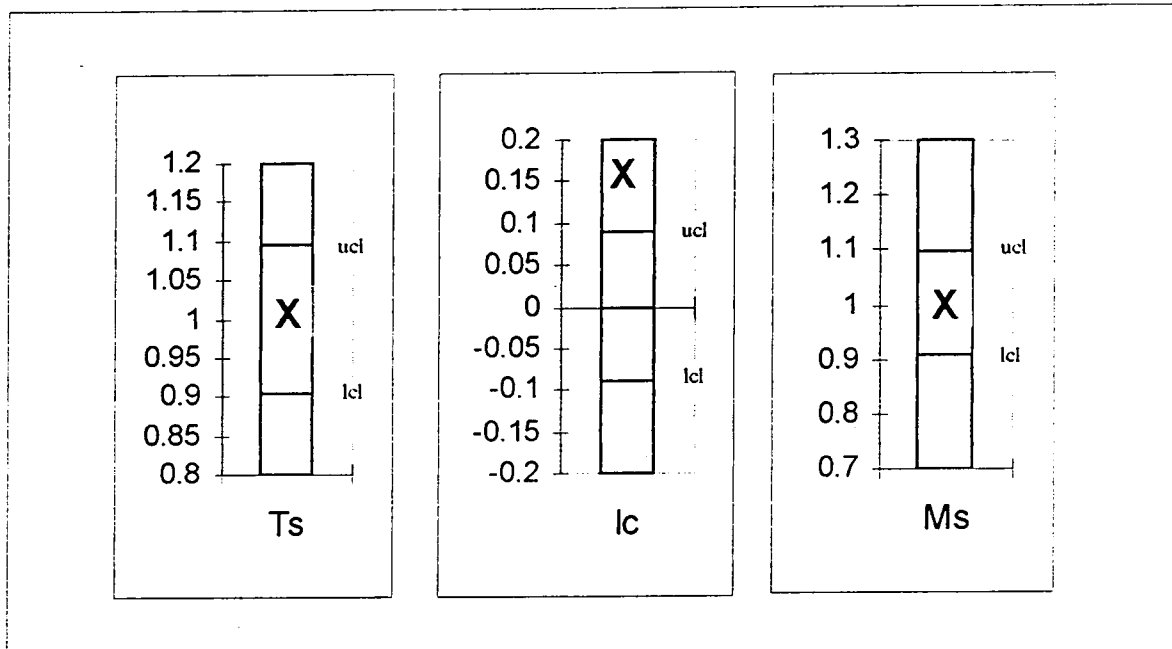


Figure 4.13 Parameter Control Chart for Profile 4

Chapter 5

Conclusions and Recommendations

5.1 Conclusions

Three conclusions can be drawn concerning the effectiveness and possible application of this technique.

1. For the profile example considered, the preprocessing step was very effective in quantifying three common deterministic sources of profile variability (initial conditions, magnitude scale, and time scale). This example was chosen to be realistic in nature but to have very well understood sources of variability. The technique developed should be equally as applicable to processes where the sources of variability are less understood.

2. The preprocessing step generates an adjusted profile which has the three deterministic sources of variability removed. This adjusted profile can then be used in an SPC analysis, thereby increasing its sensitivity to special cause variability.

3. Although not directly considered in this study, it seems as though the preprocessing step could be used for long term process improvements. It also

quantifies profiles in a form that would make it easier to correlate profile characteristics with quality, costs, and throughput.

5.2 Recommendations

In the development of this technique, several areas were identified as needing further investigation.

1. A more sophisticated, mathematical approach needs to be developed for the determination of a reference profile. It should be robust in the sense that it can be applied to profiles with many different shapes and characteristics. The proper determination of the reference profile is critical to the successfulness of this procedure. If the reference fails to capture all the characteristics of a profile, then the resulting projections will have large deviations from that reference, which could be misinterpreted as “special cause” variation.
2. Further case studies should be performed using real process data, where the forms of variability are relatively unknown. It would also be helpful to have case studies, where the variability is well understood, as an evaluation of this technique.
3. Once the procedure is applied to real process data, a study should be performed on the selection of proper control limits. Several control limit issues were identified in

this study, such as the autocorrelation of data and the use of individual control charts at points in time. These issues affect how the limits should be determined and their resulting sensitivity. This is also another area that is affected by the determination of a reference profile.

4. An investigation in the use of this procedure coupled with the use of multivariate techniques is also recommended. The technique presented in this study can classify variation as three deterministic sources with quantification. The multivariate techniques developed identify groups of variables associated with variation with some quantification. The two methods used in conjunction with each other may provide a better clue as to the sources of process variability.

5. Further study is also suggested to determine if there is a correlation between this technique and process and product characteristics such as: quality, costs, and throughput. The technique can characterize profiles by parameters which are in a more compatible form for correlation studies.

Bibliography

Duncan, A.J., Quality Control and Statistics, Richard D. Irwin, Inc., 4th ed., Homewood, Ill., 1974.

Farebrother, R.W., Linear Least Squares Computations, Marcel Dekker, Inc., New York, 1988.

Fisher, Thomas G., Batch Control Systems: Design, Application, and Implementation, Instrument Society of America, North Carolina, 1990.

Fletcher, R., Practical Methods of Optimization: Volume I Unconstrained Optimization, John Wiley & Sons, New York, 1980.

Forsythe, G.E., M. A. Malcolm, and C.B. Moler, Computer Methods for Mathematical Computations, Prentice-Hall, 1976.

Holmes, Donald., *SPC Techniques for Low-Volume Batch Processes*, Ceramic Bulletin, 69 (1990), 5, 818-821.

Kresta, James V., J.F. MacGregor, and T.E. Marlin, *Multivariate Statistical Monitoring of Process Operating Performance*, Canadian Journal of Engineering, 69 (1991), 35-47.

Lawson, Charles L., and R.J. Hanson, Solving Least Squares Problems, Prentice-Hall, Englewood, N.J., 1974.

Leitnaker, Mary G., R.D. Sanders, and Cheryl Hild, The Power of Statistical Thinking: Improving Industrial Processes, Addison-Wesley Pub. Co., Reading, Mass., 1996.

Mamzic, C.L., *Statistical Process Control*. Practical Guides for Measurement and Control. Instrument Society of America., North Carolina, 1995.

Marsh, C.E., and Tucker, T.W., *Application of SPC Techniques to Batch Units*, ISA Transactions, 30 (1991), 39-47.

Moore, Charles F., *A New Role for Engineering Process Control Focused on Improving Quality*, Competing Globally Through Customer Value: The Management of Strategic Suprasystems, Ed. Michael J. Stahl, and Gregory M. Bounds. New York: Quorum Books, 1991. 478-509.

Nelder, J.A., and R. Mead, *A Simplex Method for Function Minimization*, Computer Journal, 7 (1965), 308-313.

Nomikos, P., and MacGregor, J.F., *Monitoring of Batch Processes Using Multiway Principle Component Analysis*, AIChE Journal, 40 (1994), 1361-1375.

Nomikos, P., and MacGregor, J.F., *Multivariate SPC Charts for Monitoring Batch Processes*, Technometrics, 37 (1995), 41-59.

Oakland, John S., *Statistical Process Control: A Practical Guide*, John Wiley & Sons, New York, 1986.

Rosenof, Howard P., and Asisha Ghosh, *Batch Process Automation: Theory and Practice*, Van Nostrand Reinhold Co., New York, 1987.

Strang, G., *Linear Algebra and its Applications*, 3rd ed., Harcourt Brace Jovanovich Inc., San Diego, 1988.

Wheeler, Donald J., and David S. Chambers, *Understanding Statistical Process Control*, 2nd ed., SPC Press, Knoxville, TN., 1992.

Appendices

Appendix A

Matlab Programs Based on a Brute Force Search

The MATLAB programs found in this appendix are all based on a brute force search method that is used to solve a nonlinear least square problem. This program uses a predetermined (and adjustable) set of time parameters to adjust a profile in time. Once a profile has been time adjusted, it is projected onto the reference profile and the remaining parameters are calculated. The square error between the reference profile and the projection is then calculated. This procedure is done for each time parameter. After all the projections have been made, the parameters that produced the smallest square error are taken to be the least square solutions. This program is very time consuming and memory inefficient. It takes approximately one hour to create the control charts on a IBM, 75 Mhz, 8M RAM, 486DX personal computer.

This appendix contains two programs. One is used to create the control charts and the other is used to analyze profiles.

```
% MATLAB program that uses a increment method in a nonlinear least square analysis
% Program titled "STEP"
```

```
clear all
load refprof.txt           % Load the reference profile
load prof.txt              % Load the sample profiles
time=refprof(:,1);         % Define the time vector

figure                     % Creates a plot of the profiles
plot(time,prof)

[rows cols]=size(prof)    % Finds the size of the profile matrix

for t=1:1:rows;           % Loop to calculate the profile control limits
    apsd(t)=std(prof(t,:)); % Calculate standard deviation at time t for profiles
    uclv(t)=3*apsd(t);      % Upper control limit for the profile deviation control chart
    lclv(t)=(-3)*apsd(t);   % Lower control limit for the profile deviation control chart
end

profu=refprof(:,2)+uclv';  % Upper control limit for profile control chart
profl=refprof(:,2)+lclv';  % Lower control limit for profile control chart
zer=zeros(rows,1);         % Centerline for the deviation control chart

figure                     % Creates the profile deviation control chart
plot(time,zer)
hold on
plot(time,uclv,'-')
plot(time,lclv,'-')
title(' Profile Deviation Control Chart')
xlabel('time')
ylabel('variable')

figure                     % Creates the profile control chart.
plot(refprof(:,1),refprof(:,2))
hold on
plot(time,profu,'-')
plot(time,profl,'-')
title(' Profile Control Chart')

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Calculations used to create the projection control charts

a(:,1)=ones(rows,1);
```

```

for t=1:cols;                                % Loop to extrapolate data for the profiles
    [rows cols]=size(prof);                    % Find the size of profile matrix
    clear profe                                % Clear the expanded profile
    profe(:,2)=prof(:,t);                      % Assign profile vector
    ad=ones(400,1);                            % Create a vector of ones
    av=(profe(rows,2)+profe(rows-1,2))/2;      % Calculate average of 2 last numbers of profile
    profe(rows+1:rows+400,2)=ad*av;           % Expand the value vector of profile
    [erows ecols]=size(profe(:,2));            % Find the size of the expanded profile
    profe(:,1)=(0:erows-1)';                  % Creates an expanded time vector

    for i=1:101;                                % Loop to shift profile
        ts=.5:.01:1.5;                        % Vector of time parameters
        xi=time;                               % Renames the time vector xi
        x=ts(i)*profe(:,1);                    % Shifts the profile in time
        y=profe(:,2);                          % Picks out the expanded profile
        yi=interp1(x,y,xi);                   % Performs the interpolation
        adjprof(:,i)=yi(1:rows);              % Defines a time adjusted profile
        a(:,1)=ones(rows,1);                  % Creates a matrix "a" to solve for the remaining parameters
        a(:,2)=adjprof(:,i);                  % Pick out one profile from the adjusted profiles.
        projection(:,i)=(a*(inv(a'*a)*(a'*refprof(:,2)))); % Creates the projection.
        ls=(inv(a'*a)*(a'*refprof(:,2)));     % Calculates the remaining parameters
        para(1,i)=ts(i);                      % Stores the parameter in a matrix "para"
        para(2:3,i)=ls;
        err=refprof(:,2)-projection(:,i);      % Calculates the error
        errsqr(i)=err'*err;
    end

    [p q]=min(abs(errsq));                    % Picks out the projection with the least error
    projvar(:,t)=projection(:,q);             % Stores the projection in a matrix "projvar"
    parameters(:,t)=para(:,q);               % Stores the parameters in a matrix
end

```

```

figure                                        % Plots the projections
plot(time,projvar)

```

```

stdls1=std(parameters(1,:));                % Calculates the standard deviation for each parameter
stdls2=std(parameters(2,:));
stdls3=std(parameters(3,:));
av1=mean(parameters(1,:));                  % Calc. the average and control limits parameter 1
ul1=av1+3*stdls1;
ll1=av1-3*stdls1;
av2=mean(parameters(2,:));                  % Calc. the average and the control limits for parameter 2
ul2=av2+3*stdls2;

```

```

ll2=av2-3*stdls2;
av3=mean(parameters(3,:)); % Calc. the average and the control limits for parameter 3
ul3=av3+3*stdls3;
ll3=av3-3*stdls3;

for t=1:1:rows; % Loop to calculate the projection control limits.
sd(t)=std(projvar(t,:)); % Calculates the standard deviation of the projections.
juclv(t)=3*sd(t); % Upper control limit for the projection deviation control chart.
jlclv(t)=(-3)*sd(t); % Lower control limit for the projection deviation control chart.
end

proju=refprof(:,2)+juclv'; % Upper control limits for the projection control chart
projl=refprof(:,2)+jlclv'; % Lower control limits for the projection control chart

figure % Creates the projection deviation control chart.
plot(time,juclv,'-')
hold on
plot(time,refprof(:,2)-refprof(:,2))
plot(time,jlclv,'-')
title(' Adjusted Projection Control Chart')
xlabel('time')
ylabel('variable')

figure % Creates the projection control chart.
plot(time,proju,'-')
hold on
plot(refprof(:,1),refprof(:,2))
plot(time,projl,'-')
title(' Projection Control Chart')

```

% MATLAB program used to analyze profiles using a increment method
 % Program titled "ASTEP"

```
clear all
load refprof.txt;          % Load the reference profile
load prof.txt;             % Load the profile to be studied
time=refprof(:,1);        % Defines the time vector
dog=prof;                  % Variable used to pick out profiles from a matrix
clear prof
prof(:,2)=refprof(:,2);
```

% Programming used to analyze the profile

```
profdev=prof(:,2)-refprof(:,2); % Calculates the profile deviation
```

```
figure(3)                  % Plots on the control chart
hold on
plot(time,prof(:,2))
```

```
figure(2)                  % Plots on the control chart
hold on
plot(time,profdev)
```

% Programming used to calculate and analyze projections

```
[rows cols]=size(prof(:,2)) % Find the size of profile
ad=ones(400,1);             % Create a vector of ones
av=(prof(rows,2)+prof(rows-1,2))/2; % Calculate average of 2 last numbers of profile
prof(rows+1:rows+400,2)=ad*av; % Expand the value vector of profile
[erows ecols]=size(prof(:,2));
prof(:,1)=(0:erows-1)';
for i=1:101;                 % Loop to shift profile
    ts=.5:.01:1.5;          % Vector of time parameters
    xi=time;                 % Renames the time vector "xi"
    x=ts(i)*prof(:,1);       % Shifts the profile in time
    y=prof(:,2);             % Picks out the expanded profile
    yi=interp1(x,y,xi);      % Performs the interpolation
    adjprof(:,i)=yi(1:rows); % Defines a time adjusted profile
    a(:,1)=ones(rows,1);     % Creates a matrix "a" to solve for the remaining parameters
    a(:,2)=adjprof(:,i);
    projection(:,i)=(a*(inv(a'*a)*(a'*refprof(:,2)))); % Creates the projection.
    ls=(inv(a'*a)*(a'*refprof(:,2))); % Calculates the remaining parameters.
    para(1,i)=ts(i);          % Stores the parameters in a matrix "para"
    para(2:3,i)=ls;
```

```

err=refprof(:,2)-projection(:,i);          % Calculates the error
errsq(i)=err'*err;
end

[p q]=min(abs(errsq));                     % Picks out the projection with the least error
projvar=projection(:,q);                   % Stores the projection as projvar
parameters=para(:,q)                      % Stores the parameters as "parameters"
end

devproj=projvar-refprof(:,2);              % Calculates the deviation of the projection

figure(6)                                 % Plots the projection
hold on
plot(refprof(:,1),projvar,'b')

figure(5)                                 % Plots the projection deviation
hold on
plot(refprof(:,1),devproj)

```

Appendix B

Matlab Programs Based on a Multiple Variable Search

The MATLAB programs presented in this appendix are based on a multiple variable search. The program solves for all three least square solutions simultaneously. The search routine is based on a simplex method for minimizing a function developed by Nelder and Mead (1965). It is a built-in MATLAB m-file called "fmins". The search routine took about the same amount of time as the brute force method in Appendix A.

This appendix contains two MATLAB programs; one to create the control charts and one to analyze profiles. Each program consists of two m-files; one is the primary m-file that contains a command "fmins" that calls the second m-file which contains the search criterion for the "fmin" command.

```
% MATLAB program that uses a multivariable search in a nonlinear analysis.
% Program titled "MVS"
```

```
clear all
load refprof.txt           % Load the reference profile
load prof.txt              % Load the sample profiles
time=refprof(:,1);         % Define the time vector
[rows cols]=size(prof);    % Finds the size of the profile matrix
```

```
figure                    % Creates a plot of the profiles
plot(time,prof)
```

```
% Calculates used to create the profile control charts
```

```
for t=1:1:rows;           % Loop to calculate the control limits at time t
    apsd(t)=std(prof(t,:)); % Calculates standard deviation of profile values at time t
    uclv(t)=3*apsd(t);     % Upper control limit for the profile deviation control chart
    lclv(t)=(-3)*apsd(t);  % Lower control limit for the profile deviation control chart
end
```

```
profu=refprof(:,2)+uclv'; % Upper control limit for the profile control chart
profl=refprof(:,2)+lclv'; % Lower control limit for the profile control chart
zer=zeros(rows,1);        % Centerline for the deviation control chart
```

```
figure                    % Creates the profile deviation control chart
plot(time,zer)
hold on
plot(time,uclv,'-')
plot(time,lclv,'-')
title(' Adjusted Profile Control Chart')
xlabel('time')
ylabel('variable')
```

```
figure                    % Creates the profile control chart.
plot(refprof(:,1),refprof(:,2))
hold on
plot(time,profu,'-')
plot(time,profl,'-')
title(' Profile Control Chart')
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
% Calculations used to create the projection control charts
```

```

for t=1:cols;
    profe(1:rows,t+1)=profe(:,t);
    ad=ones(400,1);
    av=(profe(rows,t+1)+profe(rows-1,t+1))/2;
    profe(rows+1:rows+400,t+1)=ad*av;
    [erows ecols]=size(profe(:,t+1));
    profe(:,1)=(0:erows-1)';
end

global profe rows refprof t i rows
% Defines the global variables

for i=1:cols;
    % Loop to perform the least square analysis
    [x,out]=fmins('mvs1',[1 0 1],[1 1e-4]); % Calls the m-file "mvs1"
    ls(:,i)=x';
    out(10);
    % Stores the least sq. solns. in matrix (ls)

    clear adjprof
    ai=time;
    a=ls(1,i)*profe(:,1);
    y=profe(:,i+1);
    yi=interp1(a,y,ai);
    adjprof(:,2)=yi(1:rows);
    % Renames the time vector "ai"
    % Creates a time adjusted profile
    % Picks out a profile from the expanded profile matrix
    % Performs the interpolation
    % Defines a time adjusted profile

    clear a
    a(:,1)=ones(rows,1);
    a(:,2)=adjprof(:,2);
    projection(:,i)=ls(2,i)*a(:,1)+ls(3,i)*a(:,2);
    projvar(:,i)=projection(:,i);
    parameters(:,i)=ls(:,i);
    % Creates a matrix "a" to determine the projection
    % Creates the projection
    % Stores the projection in a matrix (projvar)
    % Stores the parameters in a matrix
end

figure
plot(time,projvar)
% Creates a plot of the projections

stdls1=std(parameters(1,:));
stdls2=std(parameters(2,:));
stdls3=std(parameters(3,:));
av1=mean(parameters(1,:));
ul1=av1+3*stdls1;
ll1=av1-3*stdls1;
av2=mean(parameters(2,:));
ul2=av2+3*stdls2;
ll2=av2-3*stdls2;
av3=mean(parameters(3,:));
ul3=av3+3*stdls3;
ll3=av3-3*stdls3;
% Calculates the standard deviation for the variables
% Calc. the average and control limits for parameter 1
% Calc. the average and control limits for parameter 2
% Calc. the average and control limits for parameter 3

```

```

avgls(:,1)=[av1 av2 av3]';           % Stores the averages and control limits
lsclimits(1,:)=[ul1 ll1];
lsclimits(2,:)=[ul2 ll2];
lsclimits(3,:)=[ul3 ll3];

for j=1:1:rows;                       % Loop to calculate the control limits for the projections
    sd(j)=std(projvar(j,:));           % Calculate standard deviation at time t for projections
    juclv(j)=3*sd(j);                 % Upper control limit for projection deviation control charts
    jlclv(j)=(-3)*sd(j);              % Lower control limit for projection deviation control charts
end

proju=refprof(:,2)+juclv';            % Upper control limit for the projection control chart
projl=refprof(:,2)+jlclv';            % Lower control limit for the projection control chart

figure                                % Creates the projection deviation control chart.
plot(time,juclv,'-.')
hold on
plot(time,refprof(:,2)-refprof(:,2))
plot(time,jlclv,'-.')
title(' Adjusted Projection Control Chart')
xlabel('time')
ylabel('variable')

figure                                % Creates the projection control chart.
plot(time,proju,'-.')
hold on
plot(refprof(:,1),refprof(:,2))
plot(time,projl,'-.')
title(' Projection Control Chart')

```

```
% MATLAB program that is used in the program titled "MVS".
% Program titled "MVS1"
```

```
function f = mvsl(x)
```

```
global prof rows refprof i t
```

```
% Defines the global variables
```

```
    ai=refprof(:,1);
```

```
% Renames the time vector "ai"
```

```
    a=x(1)*prof(:,1);
```

```
% Shifts the profile in time
```

```
    y=prof(:,i+1);
```

```
% Picks out a profile from the expanded profile
```

```
matrix
```

```
    yi=interp1(a,y,ai);
```

```
% Performs the interpolation
```

```
    adjprof(:,2)=yi(1:rows);
```

```
% Defines a time adjusted profile
```

```
    adjprof(:,1)=ones(rows,1);
```

```
% The error function to be minimized
```

```
f=(refprof(:,2)-(x(2)*adjprof(:,1)+x(3)*adjprof(:,2)))*(refprof(:,2)-
(x(2)*adjprof(:,1)+x(3)*adjprof(:,2))));
```

% MATLAB program used to analyze profiles by a multivariable search routine
 % Program titled "AMVS"

```
clear all
load refprof.txt          % Load the reference profile
load prof.txt             % Load the sample profiles
dog=prof;                 % Used to store the profile values
clear prof
prof(:,2)=dog(:,1);       % Defines the profile to analyze
time=refprof(:,1);        % Defines the time vector
```

% Programming used to analyze the profile

```
profdev=prof(:,2)-refprof(:,2); % Calculates the profile deviation
```

```
figure(3)                 % Plots on the profile control chart
hold on
plot(time,prof(:,2))
```

```
figure(2)                 % Plots on the profile deviation control chart
hold on
plot(time,profdev)
```

% Programming used to create and analyze the projection

```
[rows cols]=size(prof(:,2)) % Finds the size of the profile
ad=ones(400,1);             % Creates a vector of ones
av=(prof(rows,2)+prof(rows-1,2))/2; % Calculates average of 2 last values of profile
prof(rows+1:rows+400,2)=ad*av; % Expands the value vector of profile
[erows ecols]=size(prof(:,2)); % Finds the size of the expanded profile
prof(:,1)=(0:erows-1)';
```

```
global prof rows refprof time % Defines the global variables
[x,out]=fminls('amvs1',[1 0 1],[1 1.e-2]); % Calls the m-file "amvs1"
ls=x' % Stores the least sq. soln. in matrix ls
out(10);
```

```
    ai=time; % Renames the time vector
    a=ls(1)*prof(:,1); % Creates a time adjusted profile
    y=prof(:,2); % Defines the profile to be interpolated
    yi=interp1(a,y,ai); % Performs the interpolation
    adjprof(:,2)=yi(1:rows); % Defines the time adjusted profile
    clear a
    a(:,1)=ones(rows,1); % Creates a matrix (a) to determine the projection
    a(:,2)=adjprof(:,2);
    projection=ls(2)*a(:,1)+ls(3)*a(:,2); % Creates the projection
```

```
projdev=projection-refprof(:,2);
```

```
% Calculates the projection deviation
```

```
figure(5)
```

```
% Plots on the deviation projection control chart
```

```
hold on
```

```
plot(time,projdev)
```

```
figure(6)
```

```
% Plots the projection on the control chart
```

```
hold on
```

```
plot(refprof(:,1),projection)
```

```
% MATLAB program used in the program titled "AMVS"
% Program titled "AMVS1"
```

```
function f = amvs1(x)
```

```
global prof refprof n adjprof rows time
```

```
    ai=time;                % Renames the time vector
    a=x(1)*prof(:,1);      % Shifts the profile in time
    y=prof(:,2);           % Defines the profile to be interpolated
    yi=interp1(a,y,ai);    % Performs the interpolation
    adjprof(:,2)=yi(1:rows); % Defines a time adjusted profile
    adjprof(:,1)=ones(rows,1);
```

```
% The error function to be minimized
```

```
f=(refprof(:,2)-(x(2)*adjprof(:,1)+x(3)*adjprof(:,2)))*(refprof(:,2)-
(x(2)*adjprof(:,1)+x(3)*adjprof(:,2))));
```

Appendix C

Matlab Programs Based on a Single Variable Search

The MATLAB programs presented in this appendix are based on a single variable search. The program searches for a time parameter that will minimize the square error between two vectors. The search routine is based on a golden section method and a parabolic interpolation. This routine is a built-in MATLAB m-file called "f-min". This program proved to be the most time efficient, taking only a fifth of the time the other methods required.

This appendix contains two MATLAB programs; one to create the control charts and one to analyze profiles. Each program consists of two m-files; one is the primary m-file that contains a command "fmin" that calls the second m-file which contains the search criterion.

```
% MATLAB program that uses a single variable search in a nonlinear least square analysis.
% Program titled "SVS"
```

```
clear all
load refprof.txt      % Load the reference profile
load prof.txt         % Load the sample profiles
[rows cols]=size(prof) % Calculates the size of the profile matrix
time=refprof(:,1);
```

```
figure                % Creates a plot of the profiles
plot(time,prof)
```

```
% Calculations used to create the profile control charts
```

```
for t=1:1:rows;      % Loop to determine control limits at time t
    apsd(t)=std(prof(t,:)); % Calculate standard deviation of profiles values at time t
    uclv(t)=3*apsd(t);    % Upper control limit for the profile deviation control chart
    lclv(t)=(-3)*apsd(t); % Lower control limit for the profile deviation control chart
end
```

```
profu=refprof(:,2)+uclv; % Upper control limit for profile control chart
profl=refprof(:,2)+lclv; % Lower control limit for profile control chart
zer=zeros(rows,1);       % Centerline for deviation control chart
```

```
figure                % Creates the profile deviation control chart
plot(time,zer)
hold on
plot(time,uclv,'-.')
plot(time,lclv,'-.')
title(' Adjusted Profile Control Chart')
xlabel('time')
ylabel('variable')
```

```
figure                % Creates the profile control chart
plot(refprof(:,1),refprof(:,2))
hold on
plot(time,profu,'-.')
plot(time,profl,'-.')
title(' Profile Control Chart')
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

% Calculations used to create the projection control charts

```

for t=1:cols;                                % Loop to extrapolate data for all the profiles
    profe(1:rows,t+1)=prof(:,t);              % Creates a new expanded profile matrix (profe)
    ad=ones(400,1);                           % Creates a vector of ones
    av=(profe(rows,t+1)+profe(rows-1,t+1))/2; % Calculates average of last 2 numbers of profile
    profe(rows+1:rows+400,t+1)=ad*av;         % Expand the value vector of profile
    [erows ecols]=size(profe(:,t+1));         % Finds the size of the expanded profile
    profe(:,1)=(0:erows-1)';                 % Creates an expanded time vector
end

```

global profe rows refprof t n ls adjprof % Defines all global variables

```

for i=1:cols;                                % Loop to perform the least square analysis
    [x,out]=fmin('test1',(0.5),1.5,[0,1e-4]); % Calls the m-file
    ts=x;                                     % Renames the time adjustment parameter ts
    out(10);
    clear a
    a(:,1)=adjprof(:,1);                     % Creates a matrix "a" to solve for the remaining parameters
    a(:,2)=adjprof(:,2);
    projection(:,i)=ls(1,1)*a(:,1)+ls(2,1)*a(:,2); % Creates the projection
    projvar(:,i)=projection(:,i);             % Stores the projection in a matrix called "projvar"
    parameters(1,:)=ts;                       % Creates a matrix for the parameters
    parameters(2:3,:)=ls;
end

```

```

figure                                       % Creates a plot of the projections
plot(time,projvar)

```

```

stdls1=std(parameters(1,:));                % Calculates the standard deviation for the parameters
stdls2=std(parameters(2,:));
stdls3=std(parameters(3,:));
av1=mean(parameters(1,:));                  % Calc. the average and control limits for parameter 1
ul1=av1+3*stdls1;
ll1=av1-3*stdls1;
av2=mean(parameters(2,:));                  % Calc. the average and control limits for parameter 2
ul2=av2+3*stdls2;
ll2=av2-3*stdls2;
av3=mean(parameters(3,:));                  % Calc. the average and control limits for parameter 3
ul3=av3+3*stdls3;
ll3=av3-3*stdls3;
avgls(:,1)=[av1 av2 av3]';                 % Stores the averages and control limits in matrices
lsclimits(1,:)=[ul1 ll1];

```

```
lsclimits(2,:)= [ul2 ll2];
lsclimits(3,:)= [ul3 ll3];
```

```
for j=1:1:rows;           % Loop to calculate control limits for the projections
sd(j)=std(projvar(j,:));  % Calculate std at time t for projections
juclv(j)=3*sd(j);         % Upper control limit for projection deviation control chart
jlclv(j)=(-3)*sd(j);      % Lower control limit for projection deviation control chart
end
```

```
proju=refprof(:,2)+juclv; % Upper control limit for the projection control chart
projl=refprof(:,2)+jlclv; % Lower control limit for the projection control chart
```

```
figure                     % Creates the projection deviation control chart
plot(time,juclv,'-.')
hold on
plot(time,refprof(:,2)-refprof(:,2))
plot(time,jlclv,'-.')
title(' Adjusted Projection Control Chart')
xlabel('time')
ylabel('variable')
```

```
figure                     % Creates the projection control chart
plot(time,proju,'-.')
hold on
plot(refprof(:,1),refprof(:,2))
plot(time,projl,'-.')
title(' Projection Control Chart')
```

```
% MATLAB m-file that is used the the program titled "SVS"
% Program titled "SVS1"
```

```
function f = test1(x)
```

```
global profe rows refprof t i adjprof ls % Defines global variables
```

```
ai=refprof(:,1); % Renames the time vector "ai"
a=x(1)*profe(:,1); % Shifts the profile in time
y=profe(:,i+1); % Picks out a profile from the expanded profile matrix
yi=interp1(a,y,ai); % Performs the interpolation
adjprof(:,2)=yi(1:rows); % Defines a time adjusted profile
adjprof(:,1)=ones(rows,1);
ls=(inv(adjprof*adjprof)*(adjprof*refprof(:,2))); % Calculates the remaining parameters
```

```
% The error function to be minimized
```

```
f=(refprof(:,2)-(ls(1,1)*adjprof(:,1)+ls(2,1)*adjprof(:,2)))*(refprof(:,2)-
(ls(1,1)*adjprof(:,1)+ls(2,1)*adjprof(:,2)));
```

```
% Matlab Program Used to Analyze Batch Profiles
% Using a Single Variable Search
% Program titled "ASVS"
```

```
clear all
load refprof.txt           % Loads the reference profile
load prof.txt              % Loads the profile to be analyzed
profile=prof(:,25)         % Statement to pick a profile out of a matrix
[rows cols]=size(profile)  % Finds the size of the profile
time=refprof(:,1);         % Define the time vector
```

```
% Programming used to analyze the profile
```

```
profdev=profile-refprof(:,2); % Calculates the profile deviation
```

```
figure(3)                  % Plots on the control chart
hold on
plot(time,profile)
```

```
figure(2)                  % Plots on the control chart
hold on
plot(time,profdev)
```

```
% Programming used to create and analyze the projection
```

```
profe(1:rows,2)=profile;   % Creates a expanded profile
ad=ones(400,1);            % Calculates a vector of ones
av=(profe(rows,2)+profe(rows-1,2))/2; % Averages last 2 values in profile
profe(rows+1:rows+400,2)=ad*av; % Expands the value vector of the profile
[erows ecols]=size(profe(:,2)); % Finds the size of the expanded profile
profe(:,1)=(0:erows-1)';
```

```
global profe rows refprof adjprof ls
```

```
[x,out]=fmin('asvs1',.5,1.5,[0,1e-4]); % Calls the m-file "asvs1"
ts=x % Renames the time adjustment parameter ts
out(10)
a(:,1)=adjprof(:,1) % Creates a matrix "a" to solve for the remaining
parameters
a(:,2)=adjprof(:,2)
projection=ls(1,1)*a(:,1)+ls(2,1)*a(:,2); % Creates the projection
parameters(1,:)=ts; % Creates a matrix for the parameters
parameters(2:3,:)=ls;
```

```
projdev=projection-refprof(:,2); % Calculates the deviation
```

```
figure(5) % Plots on control chart  
hold on  
plot(time,projdev)
```

```
figure(6) % Plots on control chart  
hold on  
plot(time,projection)
```

```
% MATLAB m-file that is used the the program titled "ASVS"
% Program titled "ASVS1"
```

```
function f = asvs1(x)
```

```
global profe rows refprof adjprof ls      % Defines global variables
```

```
ai=refprof(:,1);           % Renames the time vector "ai"
a=x(1)*profe(:,1);        % Shifts the profile in time
y=profe(:,2);             % Picks out a profile from the expanded profile matrix
yi=interp1(a,y,ai);       % Performs the interpolation
adjprof(:,2)=yi(1:rows);  % Defines a time adjusted profile
adjprof(:,1)=ones(rows,1);
ls=(inv(adjprof*adjprof)*(adjprof*refprof(:,2))); % Calculates the remaining parameters
```

```
% The error function to be minimized
```

```
f=(refprof(:,2)-(ls(1,1)*adjprof(:,1)+ls(2,1)*adjprof(:,2)))*(refprof(:,2)-
(ls(1,1)*adjprof(:,1)+ls(2,1)*adjprof(:,2)));
```

Vita

Brett Darnell was born in Kingsport, Tennessee, on October 3, 1971. He was enrolled in the Sullivan County public school system and graduated from Sullivan North High School in June 1989. He began his freshman year at the University of Tennessee in the fall of that same year. During his undergraduate studies, he participated in the cooperative engineering program with Nuclear Fuel Services located in Erwin, Tennessee. He received a Bachelor of Science degree in Chemical Engineering in May 1994 and entered the graduate program at the University of Tennessee.