

Copyright is owned by the Author of the thesis. Permission is given for a copy to be downloaded by an individual for the purpose of research and private study only. The thesis may not be reproduced elsewhere without the permission of the Author.

UTILISING MEASURABLE UNCONTROLLABLE  
FACTORS IN PARAMETER DESIGN TO OPTIMISE  
THE RESPONSE

A THESIS PRESENTED IN PARTIAL FULFILMENT  
OF THE REQUIREMENTS FOR  
THE DEGREE OF DOCTOR OF PHILOSOPHY IN  
STATISTICS AT  
MASSEY UNIVERSITY

Megan J. Pledger

September 1998

# Abstract

This thesis looks at issues in Industrial Experimental Design and Quality Control. The first part is a review of Parameter Design and its evolution into methods of modelling the mean and variance as one system.

The second part introduces the concepts of observable and unobservable factors as an extension to the ideas of controllable and uncontrollable factors of Parameter Design. Methods will be devised to show how to choose the best settings of the controllable factors and how to move to those settings once chosen. In the last section estimates for tracking the unobservable uncontrollable factors will be devised. These will be examined to see whether they can be used to improve the monitoring of the system via control charts.

■

2/128 Sandringham Road,  
Kingsland,  
Auckland,  
New Zealand.

*To Whom It May Concern.*

This is to state that the research work carried out for my Ph.D. thesis entitled 'Utilising Measurable Uncontrollable Factors in Parameter Design to Optimise the Response' in the Statistics Department at Massey University is all my own work.

The review work in Chapter 1 was originally submitted for my Masters Degree and has been revised and updated for this thesis. Details can be found in Appendix A.

I certify that the remaining material in the thesis has not been used for any other degree.

*M J Pledge*  
*8/9/98*

# Acknowledgements

Much thanks to my supervisor, Associate Professor Hugh Morton.

Thanks to Massey University for the use of their facilities.

Thanks also to Peace Computers for the use of their computers, printers and modem.

# Contents

<b>Acknowledgements</b>	<b>iv</b>
<b>Table of Contents</b>	<b>v</b>
<b>List of Figures</b>	<b>viii</b>
<b>List of Tables</b>	<b>xii</b>
<b>1 Review</b>	<b>1</b>
1.1 Parameter Design . . . . .	1
1.2 The Design of a Product or Process . . . . .	2
1.3 Parameter Design Methodology . . . . .	2
1.4 The Method . . . . .	4
1.4.1 An Example . . . . .	4
1.5 The Analysis . . . . .	6
1.5.1 Example Continued . . . . .	6
1.5.2 Confirmation Trials . . . . .	8
1.5.3 Rationale Behind The Experimental Designs . . . . .	9
1.6 Criticism of the Method . . . . .	9
1.7 A Simpler View . . . . .	13
1.8 The Positive Aspects of Parameter Design . . . . .	16
1.9 Alternatives . . . . .	18
1.9.1 Union-Intersection Method . . . . .	18
1.9.2 Transformations and a Data Analytic Approach . . . . .	28
1.9.3 Response Surface Methods . . . . .	37
1.9.4 Split-Plot Methods. . . . .	45
1.9.5 Generalised Linear Models . . . . .	54
1.9.6 Restricted Generalised Linear Models . . . . .	66

1.9.7	Comments . . . . .	73
<b>2</b>	<b>Optimisation Under Changing Conditions</b>	<b>74</b>
2.1	A First Order Model . . . . .	75
2.1.1	Model One . . . . .	75
2.1.2	Model Two . . . . .	77
2.2	Simulation Example . . . . .	82
2.3	Complications . . . . .	88
2.4	Implicit Introduction of Uncontrollable Factors . . . . .	94
2.4.1	Example . . . . .	96
2.5	Allowing For Bias . . . . .	97
2.5.1	Simulation Example Continued . . . . .	99
2.5.2	Simulation Example Continued . . . . .	104
2.6	Bias and the Implicit Introduction of Uncontrollable Factors . . . . .	106
2.7	A Note on Notation . . . . .	108
<b>3</b>	<b>Making Optimal Changes</b>	<b>110</b>
3.1	Non-Instantaneous Adjustment . . . . .	110
3.1.1	Explicit Introduction of Unobservable Uncontrollable Factors	111
3.1.2	The Two Variable Case . . . . .	115
3.1.3	Line Integrals . . . . .	116
3.1.4	Line Integrals For the Greedy and Linear Paths . . . . .	117
3.1.5	Numerical Comparisons . . . . .	122
3.2	On Finding A Minimal Path With Calculus of Variations . . . . .	125
3.2.1	Placing Restrictions on the Solution . . . . .	130
3.2.2	The Finite Difference Algorithm . . . . .	132
3.2.3	Finding an Approximation . . . . .	134
3.2.4	Paths to Non-Minimal Points . . . . .	136
3.3	Optimal Paths with More than Two Variables . . . . .	139
3.4	Moving One Factor at a Time . . . . .	142
3.4.1	Two Factor Case . . . . .	143
3.4.2	Two or More Factors Case . . . . .	148
3.5	Moving Two Factors at a Time . . . . .	149
3.5.1	Two or More Factors Case . . . . .	150

<b>4</b>	<b>On-Line Quality Control</b>	<b>155</b>
4.1	The Distributions of $Z$ and $\epsilon$ Given $Y$ . . . . .	156
4.2	$\epsilon Y = y$ . . . . .	160
4.3	Highest Probability Estimates . . . . .	165
4.4	Evaluating the Performance of the Estimators . . . . .	174
4.5	Sensitivity to Changes in the Process . . . . .	180
4.5.1	Example . . . . .	181
4.5.2	Comments . . . . .	188
<b>5</b>	<b>Concluding Remarks</b>	<b>189</b>
<b>A</b>	<b>Review Details</b>	<b>190</b>
<b>B</b>	<b>A Publication From This Thesis.</b>	<b>191</b>
<b>C</b>	<b>Source Files For The Quadratic Program</b>	<b>202</b>
<b>D</b>	<b>Maple And Splus Commands</b>	<b>208</b>
<b>E</b>	<b>Approximation Data</b>	<b>214</b>
<b>F</b>	<b>Finite Difference Algorithm:1</b>	<b>217</b>
<b>G</b>	<b>Finite Difference Algorithm:2</b>	<b>228</b>
<b>H</b>	<b>Design Matrix for the Split Plot Data</b>	<b>239</b>
<b>I</b>	<b>Determining Minimal Transitions:1</b>	<b>242</b>
<b>J</b>	<b>Determining Minimal Transitions:2</b>	<b>248</b>
	<b>Bibliography</b>	<b>263</b>



# List of Figures

1.1	Interpretation of Factors. . . . .	3
1.2	QQPlot of coefficients for a) the analysis of the means and b) the analysis of the signal to noise ratio. . . . .	7
1.3	Plot of means for each level of the factors a) for the analysis of the mean shrinkage and b) for the analysis of the signal to noise ratio. . . . .	8
1.4	The Responses for Four Replicates of a $F2^2$ Experiment. . . . .	10
1.5	No Interaction Between $A$ and $B$ . . . . .	14
1.6	No interaction Between $A$ and $B$ . . . . .	14
1.7	An Interaction Between $A$ and $B$ . . . . .	15
1.8	An Interaction Between $A$ and $B$ . . . . .	15
1.9	The Special Areas of a Cross Plot . . . . .	30
1.10	a) Boxplot and b) Mean-Variance Plot of the Response Data . . . . .	32
1.11	Lambda Plots of a) Mean Effects and b) Log S.D. Effects after transformation . . . . .	33
1.12	a) Boxplot and b) Mean-Variance Plot of the Transformed Response Data . . . . .	34
1.13	QQPlots of a) Mean Effects and b) Log S.D. Effects after transformation . . . . .	35
1.14	Cross Plot of Transformed Data . . . . .	36
1.15	a) Contour Plot and b) Perspective Plot of the Mean Response Surface. . . . .	43
1.16	a) Contour Plot and b) Perspective Plot of the Variance Response Surface. . . . .	43
1.17	Joint Contour Plot of the Mean Response Surface and Variance Response Surface. . . . .	44
1.18	QQPlot of a) Whole Plot Effects and b) Split Plot Effects . . . . .	49
1.19	Interaction Plot of Tar and Temp . . . . .	52
1.20	a) Contour Plot and b) Perspective Plot. . . . .	53

1.21	A Graph of E.Q.D. for different values of $\alpha$ for the variance function $V(\mu) = \mu^\alpha$ . . . . .	59
1.22	QQplots of coefficients for a) Inverse link and b) Identity link c) Square root link and d) Log link. . . . .	60
1.23	QQplots of coefficients for a) Inverse link and b) Identity link c) Log link and d) Square Root link. . . . .	62
1.24	Lambda plot of the mean effects using a GLM model and variance function $V(\mu) = \mu$ . . . . .	63
1.25	QQPlot of Dispersion Effects. . . . .	70
1.26	QQPlot of Mean Effects. . . . .	72
2.1	Responses collected from simulation 1.1 and simulation 1.2. They show how accounting for the values of the observable uncontrollable factors has reduced the variance of the response. . . . .	86
2.2	A contour plot of $\hat{E}(y_i \mathbf{x}_i^{**}, \mathbf{t}_i - T)^2 - \hat{E}(y_i \mathbf{x}_i^*, \mathbf{t}_i - T)^2 = 0$ . Inside the two smaller regions $\hat{\mathbf{x}}_i^{**}$ gives the smaller $\hat{E}(y_i \mathbf{x}_i, \mathbf{t}_i - T)^2$ value. . . . .	87
2.3	A half-normal plot of the main effects and interactions for the injection moulding experiment . . . . .	91
2.4	A plot of the observable uncontrollable factor, $N$ versus the controllable factors, $A, C, D, E, G$ . . . . .	92
2.5	A plot of the response to the regression equation using the solutions in Table 2.7 for the factors and 100,000 random variates for $M$ and $N$ . . . . .	94
2.6	Responses collected in histograms for $Y$ when $T=100$ for $\mathbf{x}_i^*$ and $\mathbf{x}_i^{***}$ respectively. . . . .	100
2.7	Responses collected in histograms for $Y$ when $T=125$ for $\hat{\mathbf{x}}_i^*$ and $\hat{\mathbf{x}}_i^{***}$ respectively. . . . .	101
2.8	Responses collected in histograms for $Y$ when $T=125$ for $\hat{\mathbf{x}}_i^{****}$ when $\lambda_1 = 1$ and $\lambda_2 = 1, 2, 5$ and $10$ respectively. . . . .	105
2.9	The Solution for $C$ and $D$ for Different Values of $A$ . . . . .	108
3.1	A comparison of the directions of the greedy and straight paths . . . . .	114
3.2	A comparison of the value of $L$ for the two paths . . . . .	115
3.3	The graph of the $g$ 's versus the estimates of $g$ . . . . .	136

3.4	The optimal paths for $L(x, y) = 5x^2 + y^2$ and $L(x, y) = x^2 + 5y^2$ from (3,1) to (1,1). . . . .	138
3.5	The optimal path on $L = 5x^2 + y^2$ from (3,3) to (4,1). . . . .	138
3.6	The generation of paths that converge to the optimal path on $L = x^2 + 5y^2$ from (3,3) to (1,4). . . . .	139
3.7	The optimal path on $L = 5x^2 + y^2$ from (3,3) to (4,1) . . . . .	139
3.8	A comparison of the three paths: straight, greedy and mixture. . . .	142
3.9	A comparison of the two possible transitions and the two paths that make up each transition. . . . .	143
4.1	The graph of the value of $z_1$ and the predictions of $z_1$ when the covariance (and correlation) between $z_1$ and $z_2$ is 0.75. . . . .	171
4.2	The graph of the value of $z_2$ and the predictions of $z_2$ when the covariance (and correlation) between $z_1$ and $z_2$ is 0.75. . . . .	171
4.3	The graph of the value of $\epsilon$ and the predictions of $\epsilon$ when the covariance (and correlation) between $z_1$ and $z_2$ is 0.75. . . . .	172
4.4	The graph of the value of $z_1$ and the predictions of $z_1$ when the covariance (and correlation) between $z_1$ and $z_2$ is -0.75. . . . .	173
4.5	The graph of the value of $z_2$ and the predictions of $z_2$ when the covariance (and correlation) between $z_1$ and $z_2$ is -0.75. . . . .	173
4.6	The graph of the value of $\epsilon$ and the predictions of $\epsilon$ when the covariance (and correlation) between $z_1$ and $z_2$ is -0.75. . . . .	174
4.7	The graph of the Mean Square Error for $\hat{z}_1$ , the estimate of $z_1$ , versus the Correlation Coefficient of $z_1$ and $z_2$ . . . . .	179
4.8	The graph of the Mean Square Error for $\hat{z}_2$ , the estimate of $z_2$ , versus the Correlation Coefficient of $z_1$ and $z_2$ . . . . .	179
4.9	The graph of the Mean Square Error for $\hat{\epsilon}$ , the estimate of $\epsilon$ , versus the Correlation Coefficient of $z_1$ and $z_2$ . . . . .	180
4.10	Counts of $z_1$ and its estimate for being inside and outside the control limits. . . . .	182
4.11	Counts of $z_2$ and its estimate for being inside and outside the control limits. . . . .	183
4.12	Counts of epsilon and its estimate for being inside and outside the control limits. . . . .	184
4.13	An X-Chart of the First 50 values of $Y$ for $\rho = 0$ . . . . .	185

4.14	An X-Chart of the First 50 values of $\hat{z}_1$ for $\rho = 0$ . . . . .	185
4.15	An X-Chart of the First 50 values of $z_2$ and $\hat{z}_2$ for $\rho = 0$ . . . . .	186
4.16	An X-Chart of the First 50 values of $\epsilon$ and $\hat{\epsilon}$ for $\rho = 0$ . . . . .	186
F.1	The graph of the path found by transformout.f . . . . .	227
G.1	The optimal path on $L = x^2 + y^2 - xy$ from (3,1) to (1,1). . . . .	238

# List of Tables

1.1	Factors in the Experiment. . . . .	4
1.2	The Inner Array. . . . .	5
1.3	The Outer Array. . . . .	5
1.4	The Complete Array. . . . .	5
1.5	Design and Results for Constructed Example. . . . .	10
1.6	The Design Array and Data. . . . .	24
1.7	$\hat{D}_f$ and $\hat{D}_f^{-1}$ Estimates. . . . .	25
1.8	$\hat{D}_{f,g}$ and $\hat{D}_{f,g}^{-1}$ Estimates. . . . .	25
1.9	The $S^2(u, v)$ values for factors A and H, B and G, C and F and D and E. . . . .	26
1.10	$F2_4^{8-4}$ Design plus the Response Data . . . . .	31
1.11	RSM Design and Data. . . . .	39
1.12	Table of Coefficients for RSM Design. . . . .	40
1.13	ANOVA for RSM Design. . . . .	41
1.14	ANOVA Table for Split Plot Design. . . . .	47
1.15	The Effects for the Split Plot Example . . . . .	50
1.16	Analysis of Variance Table . . . . .	51
1.17	Extended quasi deviances for different $V(\mu)$ . . . . .	58
1.18	$EQD$ 's for the Dispersion Model. . . . .	61
1.19	$EQD$ 's for the Mean Model taking into account the Dispersion Model. . . . .	65
2.1	The experimental design and the responses for a $2^{6-1}$ factorial design in $x_1, x_2, t_1, t_2, z_1$ and $z_1$ . . . . .	83
2.2	The Maple output giving the relationship between the controllable factors, $\mathbf{x}_i$ , and the observable uncontrollable factors, $\mathbf{t}_i$ , that min- imises the variance and keeps the mean on target. . . . .	84

2.3	A summary of the simulation results. . . . .	88
2.4	The factors in the injection moulding experiment. . . . .	89
2.5	The design and the responses in the injection moulding experiment. . . . .	89
2.6	The rewritten design of the injection moulding experiment. . . . .	90
2.7	The solution giving the relationship between the controllable factors $A, B, C, D, E, F, G$ and the observable uncontrollable factor, $N$ , that minimises the variance and keeps the mean on target. . . . .	93
2.8	The equation giving the relationship between $\hat{\mathbf{x}}^{***}$ and $\mathbf{t}_i$ found using maple. . . . .	99
2.9	Responses collected in histograms for $Y$ when $T=100$ for $\mathbf{x}_i^*$ and $\mathbf{x}_i^{***}$ respectively. . . . .	100
2.10	A summary of the simulation results of the mean, variance and $MSE(T=125)$ for $Y$ . . . . .	102
2.11	A summary of the simulation results of the mean, variance and $MSE(T=125)$ for $Y$ for different values of $\lambda_2$ . . . . .	106
4.1	This table gives the sample Mean Square Error for each estimate of $z_1, z_2$ and $\epsilon$ for different Variance-Covariance matrices of the joint distribution of $z_1, z_2$ and $\epsilon$ . . . . .	170
4.2	This table gives the sample Mean Square Error and theoretical Mean Square Error for each estimate of $z_1, z_2$ and $\epsilon$ for different Variance-Covariance matrices of the joint distribution of $z_1, z_2$ and $\epsilon$ . . . . .	178

# Chapter 1

## Review

This thesis comes in two parts. The first part is a review of Parameter Design and its evolution into the joint modelling of the mean and variance. The second part extends some of the ideas of Parameter Design for use in Quality Control.

### 1.1 Parameter Design

At the end of the Second World War much of the infrastructure in Japan had been damaged or destroyed. Importance was placed in restoring and repairing the damaged infrastructure as quickly as possible. Dr Genichi Taguchi was employed to get the telephone system up and running again and his methods started their evolution during this period.

Once the infrastructure was put into place the Japanese realised that the way to rebuild the prosperity of their country was to do so through manufacturing. Producing products of high quality became an important issue for two reasons. The first was to change the poor reputation that Japanese products had before the Second World War and the second was to minimise the wastage of imported raw materials. In this climate many ideas were conceived and continue to be conceived as a means of improving quality. Dr Taguchi's methods were transferred into this environment and evolved. His ideas were influenced by his background in Engineering and Communications Theory and so they were easily passed on to other practising industrial engineers as they were written in engineering "language". Dr Taguchi's methods gave improvements which could be readily seen and this enhanced their appeal. For both of these reasons his

ideas became well known although not universally accepted in Japan.

Japan's success in the manufacturing fields came under scrutiny as other countries endeavoured to find the reasons for their success and to copy them. In the 1980's Dr Taguchi's methods were introduced to the United States of America. Since then there have been varied responses to his techniques although the underlying ideas that gave rise to Parameter Design were seen as an important breakthrough.

## 1.2 The Design of a Product or Process

Dr Taguchi divides the design of a product or process into three groups, system design, parameter design and tolerance design.

**System design** is when ideas and methods about a product or process are thought up and implemented to the stage of a working prototype. This stage involves utilising the latest technology. Quality variables are determined.

Proposed tolerances are found for them as well as parameters that are thought to affect them.

**Parameter design** is a pre-production quality control method. At this stage optimal values for the parameters are found through experimentation so that the quality variable is in some way optimised.

The last step is **Tolerance design**. Tolerances are set at the system design stage and if after the parameter design stage these tolerances are not met then action has to be taken in order to conform to these tolerances. This may involve using higher quality components or reworking inferior products. Choosing the best option involves calculating a loss function. This function is intended to represent the gain the customer gets and the loss the manufacturer has under a specific option. The options that gives the least loss is considered the best option.

System and Tolerance Design are very much the preserve of engineers however Parameter Design blends itself into the wider topics in Industrial Statistics.

## 1.3 Parameter Design Methodology

Parameter Design utilises Experimental Design and extends some of the terminology. In this new setting the parameters are split into two groups of factors, uncontrollable (noise) factors and controllable factors. Controllable



factors are those that can be controlled or that the manufacturer chooses to control during production i.e. on-line. The uncontrollable factors are ones that the manufacturer can't control or chooses not to control during production. These uncontrollable factors could include humidity in the workplace, the quality of products used or the way the customers treat the product. A factor can only belong to one of these factor sets at any one time. However sometimes the choice of which set the factor can fall into lies with the experimenters. For example humidity can be controlled during production but at great cost and so the economic choice may be to deem it to be uncontrollable. The experimentation in Parameter Design is done off-line and all the factors, controllable and uncontrollable are set at fixed levels. The uncontrollable factors are used directly or simulated and are varied with the controllable factors, using a designed experiment, and the results are noted. From this data the controllable factors are subdivided into four groups. They are

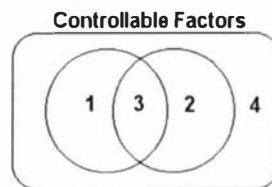


Figure 1.1: Interpretation of Factors.

- Group 1. Controllable factors that effect only the mean level of the process or product variable.
- Group 2. Controllable factors that effect only the variance of the process or product variable
- Group 3. Controllable factors that effect both the mean and variance.
- Group 4. Controllable factors that effect neither the mean nor the variance.

Group 2 and 3 members are called dispersion effects or variance control factors and group 1 and 3 members are called mean effects or target control factors.

## 1.4 The Method

The experimental designs that Dr. Taguchi advocates are orthogonal arrays which are mostly fractionated factorial designs of which a large subset are the ones described by Plackett and Burman[30] in 1946. These are resolution three designs which allow only the main effects to be estimated. One design is used for the controllable factors and is called the inner array. Another design is used for the uncontrollable factors and called the outer array. They are crossed so that for each setting of the controllable factors a complete replicate of the outer array is run. For each distinct run of the controllable factors the response mean and variance are calculated.

### 1.4.1 An Example

This example is taken from Engel's[13] paper on modeling variation. This experiment looks at the percentage shrinkage in an injection moulding process. The factors are set at two levels and are defined in Table 1.1. There are seven controllable factors and three uncontrollable factors.

Table 1.1: Factors in the Experiment.

Controllable Factors	Uncontrollable Factors
A: cycle time	M: percentage regrind
B: mould temperature	N: moisture content
C: cavity thickness	O: ambient temperature
D: holding pressure	
E: injection speed	
G: holding time	
H: gate size	

The design for the controllable factors is a  $F2_3^{7-4}$  and is shown in Table 1.2. The design for the uncontrollable factors is a  $F2_3^{3-1}$  and is shown in Table 1.3. The first design contains eight trials and the latter four and so after crossing, the entire experiment will contain 32.

Cell	Controllable Factors						
	A	B	C	D	E	G	H
1	-1	-1	-1	-1	-1	-1	-1
2	-1	-1	-1	1	1	1	1
3	-1	1	1	-1	-1	1	1
4	-1	1	1	1	1	-1	-1
5	1	-1	1	-1	1	-1	1
6	1	-1	1	1	-1	1	-1
7	1	1	-1	-1	1	1	-1
8	1	1	-1	1	-1	-1	1

Table 1.2: The Inner Array.

Cell	Uncontrollable Factors		
	M	N	●
1	-1	-1	-1
2	-1	1	1
3	1	-1	1
4	1	1	-1

Table 1.3: The Outer Array.

The data is collected and listed as in Table 1.4.

Table 1.4: The Complete Array.

i	InnerArray							OuterArray				$\bar{y}_i$	$s_i$				
	A	B	C	D	E	G	H	M	N	●	$y_{1,i}$			$y_{2,i}$	$y_{3,i}$	$y_{4,i}$	
1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	2.2	2.1	2.3	2.3	2.225	0.10
2	-1	-1	-1	1	1	1	1	-1	-1	-1	-1	0.3	2.5	2.7	0.3	1.45	1.33
3	-1	1	1	-1	-1	1	1	1	1	1	1	0.5	3.1	0.4	2.8	1.7	1.45
4	-1	1	1	1	1	1	-1	-1	-1	-1	-1	2.0	1.9	1.8	2.0	1.925	0.10
5	1	-1	1	-1	1	-1	1	1	1	1	1	3.0	3.1	3.0	3.0	3.025	0.05
6	1	-1	1	1	-1	1	-1	-1	-1	-1	-1	2.1	4.2	1.0	3.1	2.6	1.37
7	1	1	-1	-1	1	1	-1	1	1	1	1	4.0	1.9	4.6	2.2	3.175	1.33
8	1	1	-1	1	-1	-1	1	-1	-1	-1	-1	2.0	1.9	1.9	1.8	1.90	0.08

If we consider (A, B, C, D, E, G, H, M, N, ●) to be a run in the experiment then the run (-1, -1, -1, -1, -1, -1, -1, -1, -1, -1) generated the value 2.2% and (-1, -1, -1, -1, -1, -1, 1, 1) generated 2.1% for the percentage of shrinkage, etc. Suppose that our target value is 1.5% then we could say that trial two has produced the closest mean value to the target and we could set the process going with the controllable factors at the levels of this trial. On the other hand we

could look at the run with the smallest sample variance, which is trial five, work out those controllable factors that only effect the mean and use them to bring the process mean to its target value. The latter case is preferable (although not necessarily optimal) since the process will be on target and the deviations from this target value will be small.

## 1.5 The Analysis

Dr. Taguchi advocates this method of analysis. To work out those controllable factors that affect the mean value an ANOVA is done using the design matrix for the controllable factors and the mean values ( $\bar{y}_i$ ) for each run of the inner array as the response. To work out which controllable factors affect the variance of the process an ANOVA is done using the design matrix for the controllable factors and the transformed response data. The transformation that is used is called the Signal to Noise ratio (SN ratio). It has units of decibels.

The transformation depends on the intent of the experiment and engineering knowledge of the system. The three most imporant cases are when the process has a target to conform to, as in the previous example, when the process variable is to be maximised, or when the process variable is to be minimised. Note that  $i$  refers to the  $i^{th}$  distinct trial of the controllable factors.

- The first case is called “nominal is best” and the sn-ratio is

$$SN_{NB} = 10 \log_{10} \left( \frac{\bar{y}_i^2}{s_i^2} \right) \quad (1.1)$$

- The second case is called “higher is better” and the sn-ratio is

$$SN_{HB} = -10 \log_{10} \left( \frac{1}{n} \sum_{i=1}^n \frac{1}{y_i^2} \right) \quad (1.2)$$

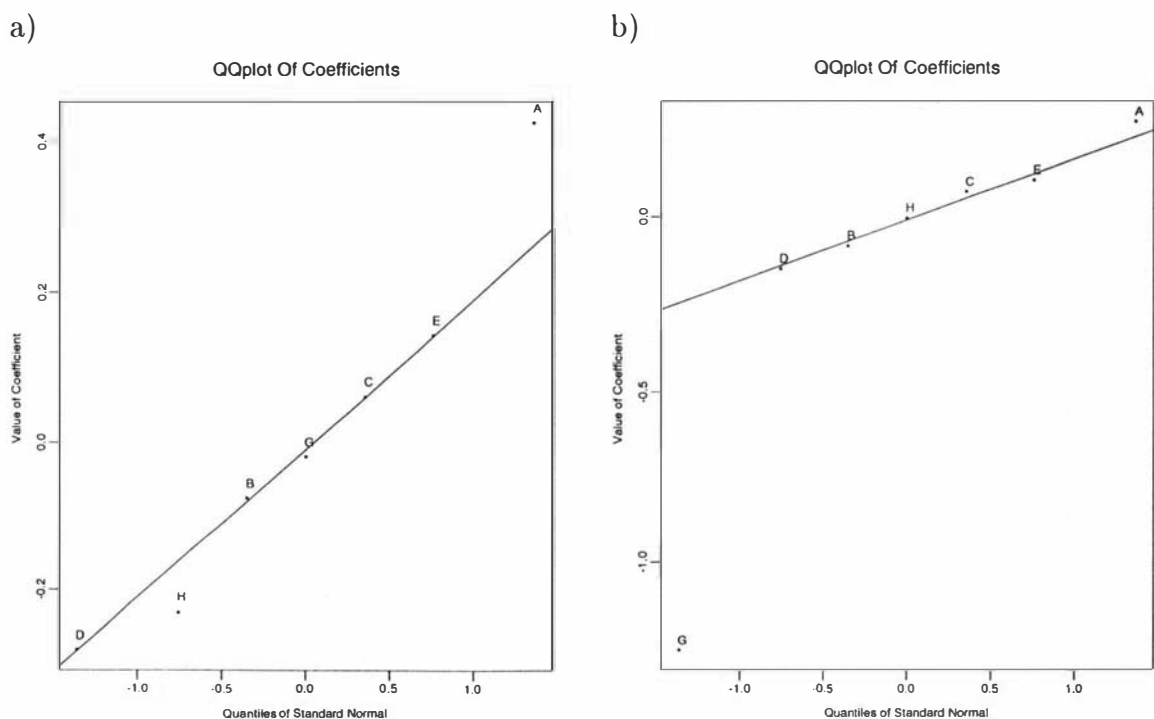
- The last case is called “lower is better” and the sn-ratio is

$$SN_{LB} = -10 \log_{10} \left( \frac{1}{n} \sum_{i=1}^n y_i^2 \right) \quad (1.3)$$

### 1.5.1 Example Continued

In order to demonstrate some of the above results I will return to the example started in section 1.4.1. In this example the percentage shrinkage of an article

from an injection moulding process has to conform to a certain target. There are eight means for the response variable and seven factors so an analysis of variance cannot be done as there are no degrees of freedom for the estimation of the error variance. A QQPlot of the coefficients of the factors found by least squares is shown in Figure 1.2a). It shows that factors A and H are factors that affect the mean level of the process since they do not lie on the line of the insignificant effects.



KEY: X represents the coefficient of the Factor X.

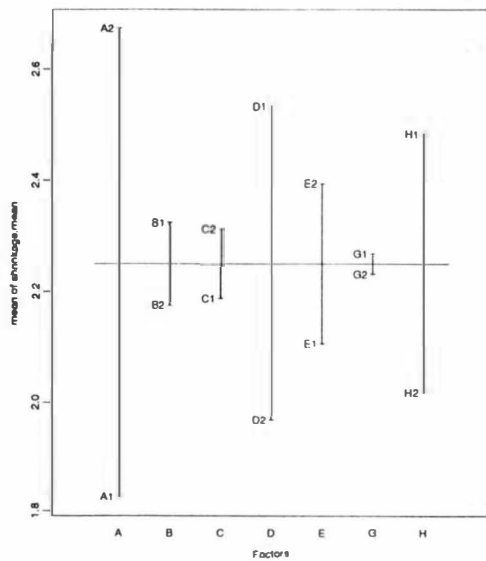
Figure 1.2: QQPlot of coefficients for a) the analysis of the means and b) the analysis of the signal to noise ratio.

Using equation 1.1 eight values for the sn-ratio were calculated. These act as the response. Since there are also seven factors an analysis of variance cannot be done as there are no degrees of freedom for estimation of the error variance. A QQPlot of the coefficients found by least squares shown in Figure 1.2b) shows that factor G is the only factor that affects the variance of the process.

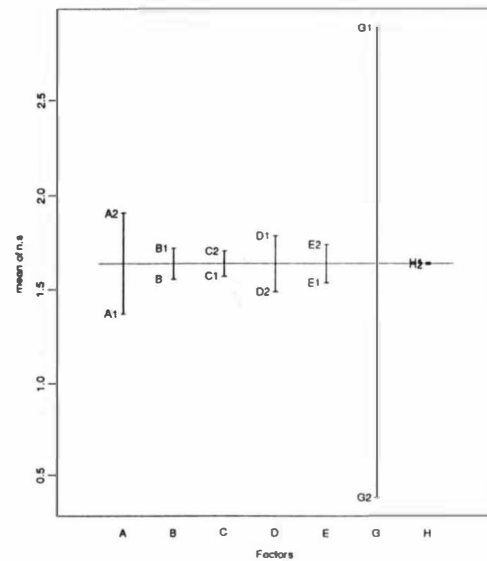
In doing an analysis of variance procedure or a QQPlot we assume that the response variable is distributed normally which in turn means that the error

distribution has a zero mean and constant variance. Constant variance is not necessarily true in some cases. Engel gave an analysis that doesn't rely on the distribution of the response variables. He plotted the averages of each factor for each level. These plots are shown in Figures 1.3a) and 1.3b). Using subjective judgement, factors A, D and H are mean control factors and possibly E. In the latter Factor G is the noise control factor.

a)



b)



KEY: X1 represents the mean for the lower level for factor X. X2 represents the upper level.

Figure 1.3: Plot of means for each level of the factors a) for the analysis of the mean shrinkage and b) for the analysis of the signal to noise ratio.

From Engel's results we would use Factor G to minimise the variance of the process (i.e. set G at its low level) and use factors A, D and H to set the process to the target shrinkage level, setting all the others at any level.

## 1.5.2 Confirmation Trials

The final step is to do a confirmation check. This involves using the derived settings to do further runs to check that these new responses match the expected responses from the model.

### 1.5.3 Rationale Behind The Experimental Designs

The Inner and Outer arrays are both level 3 Designs. In these designs the main effects are aliased with the two factor interactions (and higher interactions) but not amongst themselves. Crossing the inner and outer arrays allows for the estimation of the Control $\times$ Noise ( $C \times N$ ) interactions. However the  $C \times C$  and  $N \times N$  interactions are confounded. The approach that advocates [25] of Dr. Taguchi's method of Parameter Design use, is to choose the quality variable and the SN ratio so that the  $C \times C$  are absent or small. However this requires engineering knowledge of the system which may not always be the case. The confirmation trials allow for the checking of large  $C \times C$  interactions. If these interactions are present then Dr Taguchi advocates that a new SN ratio or a new quality response should be chosen and the process iterated.

The advocates of Dr. Taguchi's method of Parameter Design say that by choosing the SN ratio and quality variable so that there are no  $C \times C$  interactions means that

- the number of runs required to estimate the parameters is reduced
- large factor sets can be split up into smaller sets allowing different teams to simultaneously do experimentation. This shortens research and development time.
- the optimum settings in the lab will translate into the factory and the field. If there are strong  $C \times C$  interactions then there are likely to be interactions between the control variables and the unobserved factor of 'place'. If this is so then the optimum found in the lab won't translate into the field or factory.

## 1.6 Criticism of the Method

There are two major criticisms of the Taguchi method of Parameter Design. The first is with the SN ratio and the second is with the concept of interactions.

The SN ratio is a measure used to find the controllable factors that effect the variance of a process i.e. the Variance Control Factors. However Box [7] showed that the SN-ratio may not always pick up the variance control factors. For example - after experimentation the responses generated by each distinct run of

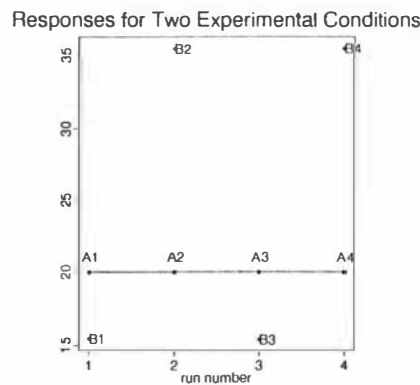
the controllable factors are condensed into a single value, the SN ratio, which is analysed by ANOVA or some related technique.

In the constructed example below consider this  $F2^2$  factorial design with factors C and D that has been replicated four times and outlined in Table 1.5

Table 1.5: Design and Results for Constructed Example.

trial no	C	D	$y_1$	$y_2$	$y_3$	$y_4$	$s_i^2$	SN Ratio
1	-1	-1	20	20	20	20	0	26.0206
2	1	-1	15.4246	35.4246	15.4246	35.4246	133.3333	26.0206
3	-1	1	20	20	20	20	0	26.0206
4	1	1	15.4246	35.4246	15.4246	35.4246	133.4333	26.0206

and represented graphically in Figure 1.4. Suppose we want to maximise the target value then we would use equation 1.2 of Section 1.5 to get the SN ratios outlined in Table 1.5 .



KEY: The Points A represent the four responses found from trial 1 ( or equivalently trial 3 ). The Points B represent the four responses found from trial 2 ( or equivalently trial 4 ).

Figure 1.4: The Responses for Four Replicates of a  $F2^2$  Experiment.

Doing an ANOVA on the SN ratios would give the result that none of the factors are variance control factors for the process as the SN ratios are all equal.

However, from observation, at the low level of C, the variances are both zero while at the high level the variances are both 133.3333. This tells us that C is a factor that influences the variance of the process. In the graph it is clear that the



variance of runs A and B are very different. The analysis that Taguchi advises has failed to pick this up.

Box [7] also showed that there were problems with  $SN_{LB}$  (see equation 1.3). This is used to find the controllable factors that affect the variance of the process and is made up of

$$\sum_{i=1}^n y_i^2$$

and this follows the formula

$$\sum_{i=1}^n y_i^2 = n\bar{y}^2 + (n-1)s^2. \quad (1.4)$$

This means location effects are being confounded with dispersion effects i.e. if factor A is only a significant location effect it may be found to be a significant dispersion effect when using  $SN_{LB}$ .

These designs, the inner and outer arrays, are both highly fractionated which means only main effects can be modelled. If there are any interactions amongst the factors in each group their effect is going to be incorporated into the aliased main effects. By crossing the inner and outer arrays together this new design allows the Noise×Control interactions to be estimated. The Noise×Noise interactions are of no importance to making the process insensitive to noise. This leaves the C×C interactions. Taguchi advocated that only the responses that cause the control factors to be additive in the range of interest should be studied. He advocated that engineers should use their knowledge of the process in order to choose the right quality response to use. However there is no guarantee that there is such a quality response or if there is one that it is known. A chosen quality response may be assumed to be linear but may not be linear in practice and then the method will fail.

Depending on the intent of the experiment and the interpretation of the experimenter a factor could be uncontrollable or controllable. To assume that this factor will interact with the control factors when it is a noise factor and not interact with any control factors when it is one itself is not sound. Jerome Sacks and William Welch[25] are quoted as saying

Taguchi's main motivation for ignoring interactions between control factors appears to be economy of experimental effort rather than any assurance it is safe to do so.

Even when ignoring the  $C \times C$  interactions the experiment can be quite large. Suppose there are  $M$  control factors and  $N$  noise factors then an experiment will consist of at least  $M \times N$  trials. For even moderate  $M$  and  $N$  this is a large number of experiments.

Dr Taguchi says [25] that

..the objective of parameter design is very different from a pure scientific study. The goal in parameter design is not to characterize the system but to achieve robust function. Pure science strives to discover the causal relationships and to understand the mechanics of how things happen. Engineering, however, strives to to achieve the result needed to satisfy the customer. Moreover, cost and time are very important issues for engineers. Science is to explain nature while engineering is to utilize nature.

This is understandable if the product is only going to be made for a short time. However if the production run is going to be long then an iterative method of experimentation is best. This is where the results of the last experiment are used to point towards new areas of investigation and experimentation. The best decisions are made in the presence of maximum information combined with engineering experience.

There are also some minor quibbles regarding some of the dangers in some of the analyses that were not acknowledged in outlining the method. They are associated with the analysis of experimental design in general and are worth recording because of their wider use and mis-use.

In highly fractionated designs there are often not enough degrees of freedom to estimate the variance of the error. Taguchi advocated pooling the smallest effects for use as the error variance. This often leads to non-significant effects being labelled significant. Taguchi explained that including non-significant effects into the solution does not affect the result, i.e. a non-significant effect at a high or low level will give the same estimated value for the response. However Box[25] argued that it is necessary to know which factors are truly significant and those which are not as this will give a clearer understanding of the system and will provide avenues of exploration for the next stage of experimentation.

In an experiment the concept of randomisation has to be adhered to. Problems arise when the experiment is run, unintentionally, as a split plot design. This

occurs when for each distinct run of the controllable factors all the different runs of the uncontrollable factors are done (or vice versa). In such cases the reasoning behind this has been economic in that it is often cheaper to change the settings of the uncontrollable factors than the controllable factors or vice versa but not both sets together. The correct way to do the experiment is to list the  $M \times N$  runs and then choose the ordering by randomly picking runs from this list. Of course the original experiment could be re-analysed taking into account the split-plot effect. The reasoning behind the SN ratios is also different from the way statisticians are taught. Parameter design puts forward a transformation before the data is collected and a confirmation trial is done to see if the model fits. Statisticians work in the opposite direction. They collect the data but let the data show what, if any, transformation need be done.

The first method is a cookbook approach. At each step there is an appropriate action that does not rely on an statistical experience e.g. viewing residual plots to check for outliers. This recipe style of presentation is easy to follow and so many experiments were done. In a lot of cases this did lead to improvements, however when things went wrong the 'cook' could be left not knowing what went wrong nor what to do next.

## 1.7 A Simpler View

The method of Parameter Design proposed by Dr. Taguchi, under some conditions, is more complicated than is needed. Rather than crossing arrays and using SN-ratios a classical experimental design can be run. In Parameter Design the way that controllable factors act on the quality variable have been given names i.e. mean control factors, variance control factors. By observing the way the controllable factors interact with the uncontrollable factors it can be seen that there is a direct connection between the Parameter Design terminology and classical Experimental Design terminology.

Consider the simplest experiment - two factors one a controllable factor  $A$  and the other an uncontrollable factor  $B$ . In the usual way of Parameter Design the experiment is done off-line and factors  $A$  and  $B$  are set at fixed levels. From the resulting experiment we can see how these factors interact. Consider these interaction graphs

In Figure 1.5 there is no interaction between  $A$  and  $B$ . The variance of  $Y$ , the quality variable, at the low level of  $A$  is the same as at the high level of  $A$ . The mean of  $Y$  at the low level of  $A$  is the same as at the high level of  $A$ . By connecting the low levels of  $B$  with a line and the high levels of  $B$  with a line we can see that the mean and variance of  $Y$  would not be predicted to change at points between the low and high level of  $A$ . Therefore  $A$ , by definition, would be neither a mean nor a variance control factor i.e. adjusting  $A$  doesn't affect the mean or the variance of  $Y$ . In classical experimental design we would say factor  $A$  does not have a main effect or an interaction with  $B$ .

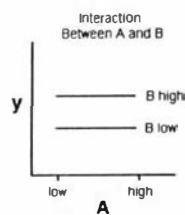


Figure 1.5: No Interaction Between  $A$  and  $B$

In Figure 1.6 there is an increase in  $Y$  between the high and low levels of  $A$ . The variance of  $Y$  at the low level of  $A$  is the same as at the high level of  $A$ . However the mean of  $Y$  at the low level of  $A$  is less than at the high level of  $A$ . Connecting the levels of  $B$ , as before, we can see that at points between the high and low levels of  $A$  there is a change in mean of  $Y$  but no change in variance. Therefore  $A$  would be a mean but not a variance control factor i.e. adjusting  $A$  affects the mean of  $Y$  but not the variance of  $Y$ . In classical Experimental Design we would say  $A$  has a main effect but that it has no interaction with  $B$ .

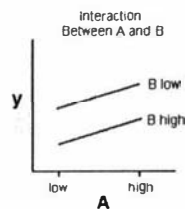


Figure 1.6: No interaction Between  $A$  and  $B$

In Figure 1.7 there is an interaction between  $A$  and  $B$ . The variance of  $Y$  at the low level of  $A$  is less than at the high level of  $A$ . However the mean of  $Y$  at the

low level of  $A$  is the same as at the high level of  $A$ . Connecting the levels of  $B$ , as before, we can see that at points between the high and low levels of  $A$  there is no change in mean but there is a change in variance. Therefore  $A$  would be a variance control factor but not a mean control factor i.e. adjusting  $A$  would affect the variance of the response but not the mean. In classical Experimental Design we would say that  $A$  had no main effect but has an interaction with  $B$ .

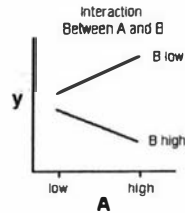


Figure 1.7: An Interaction Between  $A$  and  $B$

In Figure 1.8 there is also an interaction between  $A$  and  $B$ . The variance of  $Y$  at the low level of  $A$  is less than that at the high level of  $A$ . The mean of  $Y$  at the low level of  $A$  is less than at the high level of  $A$ . Connecting the levels of  $B$ , as before, we can see that at points between the levels of  $A$  there is a change in mean and a change in variance. Therefore  $A$  would be a mean and variance control factor i.e. adjusting  $A$  would affect the mean and variance of  $Y$ . In classical Experimental Design we would say that  $A$  has both a main effect and an interaction with  $B$ .

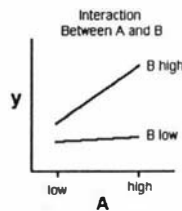


Figure 1.8: An Interaction Between  $A$  and  $B$

To determine those factors that are mean and/or variance control factors all that is necessary is to have an experimental design that allows for main effects and interactions between control and noise factors. The inner and outer arrays of Parameter Design allow for this as do many classical experimental designs. The latter are usually smaller and also allow for interactions and higher order

relations between control factors and between noise factors. The Split Plot and Response Surface methods outlined in the next section use this rationale.

However for testing procedures it introduces the assumption of a constant error variance which may not always be seen in practice.

Most of the current research has focused more closely on the idea of modelling the mean and variance in terms of the control factors alone. The noise factors may be introduced into the experimental design but it is not an absolute requirement. This is handy when the experimenter

- 1) does not know all or many of the possible noise factors or
- 2) what levels the noise factors should be set at if they are known.

This joint modelling of the mean and variance requires more latitude in the specification of their distributions. The methods seen in the next chapters allow for this requirement.

## 1.8 The Positive Aspects of Parameter Design

According to Pignatiello and Ramberg[32] the main triumph of of Dr Taguchi's method of parameter design is that is has been actively promoted. Although Box [25] showed that some of Dr Taguchi's ideas had been available in the West

1) the othogonal arrays were very similar to the fractionated factorials of Plackett and Burman[30] and

2) that the idea of uncontrollable variables had also <sup>been</sup> conceptualised by Michaels[22] <sup>1</sup>. However this knowledge had never been put together to give a method of quality control for use in the workplace.

People at AT&T, Ford, ITT and Xerox and The American Supplier Institute have put some effort into promoting Dr Taguchi's method in the United States of America, the latter supplying articles for manufacturing magazines such as Process Engineering[17], Engineering[2] and Metallurgia[3]. These ideas also have been spread through seminars and courses. This has had two impacts. A wider range of people, managers as well as engineers, have been introduced to quality engineering and using statistical techniques to make improvements in quality. It has also increased the awareness of the variances role in quality improvements.

---

<sup>1</sup>In his paper Michaels talked about the importance of knowing about the interactions between the product and the environment (ie Control  $\times$  Noise Interactions) so that the quality of the product would be competitive in all environments.

The publicity has led to experiments being done and improvements in quality have been seen as a result. However these improvements have not always been optimal.

Unfortunately some of the teaching in these seminars and courses has been poor. Teachers have given incorrect information or omitted to outline the appropriate assumptions behind the method. This has led to opposition and in some cases very strong opposition to the 'Taguchi Method'. A polarisation occurred as groups formed into those that believed the Taguchi method of Parameter Design was the be-all and end-all in quality improvement and another group that have been highly critical and in some cases antagonistic. Bob Pease writing in *Electronic Design* [27] [29] had a very negative opinion of the Taguchi Methods partly due to being introduced to it in a condescending manner and with unrealistic examples. For example during the seminar Mr Pease noted that

It looked to me as if somebody were laying down some funny data to make it easy for us to analyze. That made me very suspicious, so I was on my toes looking for any other suspicious statements.

The lecturer then belittled the operator of the machines - 'Joe thought he knew how to run the machines better than anybody else, but the Taguchi method found an optimum operating point that Joe never thought of ...'. Again I was suspicious, because that might be true, but maybe Joe knows something that the computers don't. For example, it's easy to prove, by math or computer, that in many cases you get the best acceleration and performance from a car if you slip the clutch a lot. All very true until the clutch burns out.'

In the follow-up article Mr Pease described an example taken from Dr Taguchi's *Introduction to Off-line Quality Control* [36] and showed that it had been presented poorly and was wrong.

The comments Bob Pease made generated much feedback[28] from both the supporters and opponents of the 'Taguchi Method'.

However there were a some academics and practitioners who realised that the concept of modelling both the mean and variance and the ideas of off-line quality control were important developments in quality improvement. They set out to build on and improve Dr. Taguchi's methods.

## 1.9 Alternatives

Several methods have been devised to improve upon Dr. Taguchi's method of Parameter Design. Six approaches taken by various authors will be outlined here. The methods are the work of the acknowledged authors. The examples are my own.

### 1.9.1 Union-Intersection Method

This method devised by Ghosh and Duh[16] aims to produce a set of runs that produce the least variance in a response variable. For each controllable factor a sample variance is produced for the high and low levels of that factor. An F test is done to see whether the variances at each level are significantly different. If a variance is significantly different at one of its levels then the level giving least variance is preferred. If there is no significant difference then no particular level is preferred.

For an interaction between two factors this method looks at whether the variance of the factors is better when they are both at either high or low levels compared to when they are at different levels. An F test is done to make the comparison. When there is no significant difference no level is preferable. If there is a significant difference between the two factors set at different levels, compared to when they are set at the same levels, then the individual variances, at all four factor levels, are compared. Multiple rules are given for which setting or settings are preferred.

When all these preferences or decisions from the test have been compiled they are sorted into sets of statements that agree and statements that disagree. These are recombined by rules based on unions and intersections to give a set of runs with minimal variance.

The method makes the assumption that any point in the experimental region will be unbiased to the target value.

#### Definitions

The level combinations of control factors or a run is denoted by

$t_i = (t_{i,1}, \dots, t_{i,f}, \dots, t_{i,m})$ , where  $t_{i,f}$  is the level of the control factor  $f$  on the  $i^{\text{th}}$  run,  $t_{i,f} = 0, 1$ ,  $f = 1, \dots, m$  and  $i = 1, \dots, n$  i.e. there are  $m$  factors and  $n$  runs.



The design matrix,  $\mathbf{D}$ , contains the  $n$  runs of which the  $i^{\text{th}}$  row is  $\mathbf{t}_i$ . For the  $i^{\text{th}}$  run there are  $r$  replicates denoted  $(y_{i,1}, \dots, y_{i,r})$ . The model assumed for the data is the general linear model with an error structure given by a diagonal matrix not necessarily  $\Sigma = \sigma^2 \mathbf{I}$ . The parameter  $\sigma_i^2$  is the  $i^{\text{th}}$  diagonal element of  $\Sigma$  and is called the dispersion of run  $i$ .

### Example

Consider this experiment where there are  $m = 2$  factors,  $f$  and  $g$  and  $n = 4$  runs with design matrix

$$D = \begin{pmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \\ 1 & 1 \end{pmatrix}$$

Where the first column represents the levels of factors  $f$  and the second column the levels of factor  $g$ . The second run is  $\mathbf{t}_2 = (1, 0)$  with  $f$  at its high level and  $g$  at its low level. ■

For factors  $f$  and  $g$ ,  $f, g \in \{1, \dots, m\}$ ,  $f \neq g$

$$\delta_{i,u}^f = \begin{cases} 1 & \text{if the level of } f \text{ in the } i^{\text{th}} \text{ run is } u \\ 0 & \text{otherwise} \end{cases}$$

$$\delta_{i,u,v}^{f,g} = \begin{cases} 1 & \text{if the levels of } f \text{ and } g \text{ in the } i^{\text{th}} \text{ run are } u \text{ and } v \\ 0 & \text{otherwise} \end{cases}$$

### Example Continued

So  $\delta_{1,0}^f = 1$  as  $f$  is set at level 0 on the first run and  $\delta_{4,0}^g = 0$  as on the fourth run  $g$  is not set at 0.  $\delta_{1,0,1}^{f,g} = 0$  as  $f$  and  $g$  are not set at 0 and 1 respectively in the first run. ■

Now define

$$\sum_{i=1}^n \delta_{i,u}^f = n_f(u)$$

and

$$\sum_{i=1}^n \delta_{i,u,v}^{f,g} = n_{f,g}(u, v)$$

so that  $n_f(u)$  is just the number of times that factor  $f$  is set at level  $u$  in the design matrix. and  $n_{f,g}(u, v)$  is just the number of times that factors  $f$  and  $g$  are set at levels  $u$  and  $v$  in the design matrix,

$$n_f(u) \sigma_f^2(u) = \sum_{i=1}^n \delta_{i,u}^f \sigma_i^2 \quad (1.5)$$

$$n_{f,g}(u, v) \sigma_{f,g}^2(u, v) = \sum_{i=1}^n \delta_{i,u,v}^{f,g} \sigma_i^2 \quad (1.6)$$

and  $\sigma_f^2(u)$  and  $\sigma_{f,g}^2(u, v)$  are the average variances for the specified factor at the specified levels.

The dispersion main effects of factor  $f$  for  $f = 1, \dots, m$  is defined as

$$D_f = \frac{\sigma_f^2(1)}{\sigma_f^2(0)} \quad (1.7)$$

This has an inverse denoted by

$$D_f^{-1} = \frac{\sigma_f^2(0)}{\sigma_f^2(1)} \quad (1.8)$$

The dispersion interaction effect for factors  $f$  and  $g$ ,  $f, g = 1, \dots, m$ ,  $f \neq g$  is defined as

$$D_{f,g} = \frac{\sum_{u=0}^1 n_{f,g}(u, u) \sigma_{f,g}^2(u, u) / \sum_{u=0}^1 n_{f,g}(u, u)}{\sum_{u=0}^1 \sum_{v=0, u \neq v}^1 n_{f,g}(u, v) \sigma_{f,g}^2(u, v) / \sum_{u=0}^1 \sum_{v=0, u \neq v}^1 n_{f,g}(u, v)} \quad (1.9)$$

In practice  $S_i^2$  is substituted for  $\sigma_i^2$  in equations 1.5 and 1.6 to provide  $S_f^2(u)$  and  $S_{f,g}^2(u, v)$ . These are used to get  $\hat{D}_{f,g}$  and  $\hat{D}_f$  from equations 1.9 and 1.7. The critical values,  $\Delta$ , for  $\hat{D}_f$  and  $\hat{D}_f^{-1}$  are F statistics.  $S_i^2$  has  $r - 1$  degrees of freedom. The numerator of  $\hat{D}_f$  is made up from  $a$  of the  $S_i^2$ 's where  $a = n_f(1)$  so the degrees of freedom in the numerator is  $a \times (r - 1)$ . The denominator of  $\hat{D}_f$  is made up from  $b$  of the  $S_i^2$ 's where  $b = n_f(0)$  so the degrees of freedom in the denominator is  $b \times (r - 1)$ . So define for  $\hat{D}_f$

$$\Delta_f^{(1)} = F_{(1-\alpha), a \times (r-1), b \times (r-1)}$$

and similarly for  $\hat{D}_f^{-1}$ ,

$$\Delta_f^{(2)} = F_{(1-\alpha), b \times (r-1), a \times (r-1)}.$$

By similar arguments define for  $\hat{D}_{f,g}$ ,

$$\Delta_{f,g}^{(1)} = F_{(1-\alpha), c \times (r-1), d \times (r-1)}$$

and for  $\hat{D}_{f,g}^{-1}$ ,

$$\Delta_{f,g}^{(2)} = F_{(1-\alpha), d \times (r-1), c \times (r-1)}$$

where  $c = n_{f,g}(0, 0) + n_{f,g}(1, 1)$  and  $d = n_{f,g}(0, 1) + n_{f,g}(1, 0)$ .

## Method

A run,  $t_i$ , is a  $1 \times m$  vector each place corresponding to either the high or low level of one of the  $m$  factors. A set of decisions is made about the value of each the places, i.e. a 0 or a 1, corresponding to the low or high level. The decisions may be ambiguous.

### The decisions made on the dispersion main effects.

- If  $\hat{D}_f \geq 1$  and  $\hat{D}_f > \Delta_f^{(1)}$  then the place in the run vector corresponding to the factor  $f$  is 0.
- If  $\hat{D}_f^{-1} \geq 1$  and  $\hat{D}_f > \Delta_f^{(2)}$  then the place in the run vector corresponding to the factor  $f$  is 1.
- In all other cases there is no decision.

### The decisions made on the dispersion interactions.

- If  $\hat{D}_{f,g} \geq 1$  and  $\hat{D}_{f,g} > \Delta_{f,g}^{(1)}$  then we consider  $\text{Min}[ S_{f,g}^2(0, 0), S_{f,g}^2(1, 1) ]$ .
  - If  $\text{Min}[ S_{f,g}^2(0, 0), S_{f,g}^2(1, 1) ] > \text{Min}[ S_{f,g}^2(0, 1), S_{f,g}^2(1, 0) ]$  then the places corresponding to  $f$  and  $g$  are not equal otherwise
  - If  $\text{Min}[ S_{f,g}^2(0, 0), S_{f,g}^2(1, 1) ] \leq \text{Min}[ S_{f,g}^2(0, 1), S_{f,g}^2(1, 0) ]$ 
    - \* if  $\text{Min}[ S_{f,g}^2(0, 0), S_{f,g}^2(1, 1) ] = S_{f,g}^2(0, 0)$  then there are two decisions
      1. places  $f$  and  $g$  are not equal and
      2. places  $f$  and  $g$  both equal 0.
    - \* If  $\text{Min}[ S_{f,g}^2(0, 0), S_{f,g}^2(1, 1) ] = S_{f,g}^2(1, 1)$  instead then
      1. places  $f$  and  $g$  are not equal and
      2. places  $f$  and  $g$  both equal 1.

- If  $\hat{D}_{f,g}^{-1} \geq 1$  and  $\hat{D}_{f,g} > \Delta_{f,g}^{(2)}$  then we consider  $\text{Min}[ S_{f,g}^2(0, 1), S_{f,g}^2(1, 0) ]$ .
  - If  $\text{Min}[ S_{f,g}^2(0, 1), S_{f,g}^2(1, 0) ] > \text{Min}[ S_{f,g}^2(0, 0), S_{f,g}^2(1, 1) ]$  then the places corresponding to  $f$  and  $g$  are equal otherwise
  - If  $\text{Min}[ S_{f,g}^2(0, 1), S_{f,g}^2(1, 0) ] \leq \text{Min}[ S_{f,g}^2(0, 0), S_{f,g}^2(1, 1) ]$ 
    - \* if  $\text{Min}[ S_{f,g}^2(0, 1), S_{f,g}^2(1, 0) ] = S_{f,g}^2(0, 1)$  then there are two decisions
      1. places  $f$  and  $g$  are equal and
      2. place  $f$  equals 0 and place  $g$  equals 1.
    - \* If  $\text{Min}[ S_{f,g}^2(0, 1), S_{f,g}^2(1, 0) ] = S_{f,g}^2(1, 0)$  instead then
      1. places  $f$  and  $g$  are equal and
      2. place  $f$  equals 1 and  $g$  equals 0.
- In all other cases there is no decision.

### Combining Decisions

After working out the possible values for each position in the run vector there may be some ambiguities. Two groups of sets are defined. The first are sets of disagreements which contain all the ambiguities and the second is the set of agreements. The group of disagreements are put into sets so that a statement in a set only disagrees with the members in its set and not with a member in another set of disagreements i.e. two set of disagreements do not disagree with each other. For example we might have this group of disagreements

- place 1 does not equal place 2,
- place 1 equal 0,
- place 2 equals 0,
- place 3 does not equal place 4,
- place 3 equal 0 and
- place 4 equals 0

then we would get two sets of disagreements namely

- $G_1 = \{ \text{place 1 does not equal place 2, place 1 equals 0, place 2 equals 0} \}$  and

- $G_2 = \{\text{place 3 does not equal place 4, place 3 equals 0, place 4 equals 0}\}$ .

i.e. sets  $G_1$  and  $G_2$  do not disagree.

For all the maximal subsets of the disagreements we find the sets that are in agreement. For  $G_1$  the subsets would be  $\{\{\text{place 1 does not equal place 2, place 1 equals 0}\}, \{\text{place 1 does not equal place 2, place 2 equals 0}\}, \{\text{place 1 equals 0, place 2 equals 0}\}$ . Finally each subset from each subset of maximal disagreements and the agreement set is intersected to give a sequence of runs that have minimal variance. For example if an agreement set was  $G_3 = \{\text{place 5 equals 0}\}$  then  $G_1$  gives subsets

- $(0,1,x,x,x), (1,0,x,x,x), (0,0,x,x,x)$

and  $G_2$  gives

- $(x,x,0,1,x), (x,x,1,0,x), (x,x,0,0,x)$

and  $G_3$

- $(x,x,x,x,0)$

where the x's represent either a 1 or a 0.

Taking the intersection of all these sets gives this set of runs

$$\{ (0, 1, 0, 1, 0) \quad (0, 1, 1, 0, 0) \quad (0, 1, 0, 0, 0) \\ (1, 0, 0, 1, 0) \quad (1, 0, 1, 0, 0) \quad (1, 0, 0, 0, 0) \\ (0, 0, 0, 1, 0) \quad (0, 0, 1, 0, 0) \quad (0, 0, 0, 0, 0) \}$$

### Example

This section gives an example of the previous method at work. A  $F 2_4^{8-4}$  design was used to construct an experiment. The data was normally distributed with factors A and C being dispersion effects. The overall mean was 125. There were no mean effects and this complies with the condition that any design point in the experimental region is unbiased for the target value of 125. The variances were

A	C	Variance
-1	-1	4
1	-1	11
-1	1	12
1	1	33

Six replicates were taken at each design point. The data is outlined in Table 1.6. Note the data has been rounded to 1 decimal place and the low level is denoted by 0 and the high level by 1.

$i$	DesignArray								Data						$S_i^2$
1	0	0	0	0	0	0	0	0	125.3	125.6	122.3	124.3	127.6	123.1	3.60
2	1	0	0	0	1	1	1	0	127.5	123.4	118.8	122.1	127.3	115.9	21.22
3	0	1	0	0	1	1	0	1	122.4	125.6	127.5	126.6	128.0	129.8	6.43
4	1	1	0	0	0	0	1	1	124.9	127.8	118.6	128.0	122.5	130.8	19.39
5	0	0	1	0	1	0	1	1	126.6	126.8	119.5	117.6	131.2	121.8	26.53
6	1	0	1	0	0	1	0	1	128.9	125.5	117.5	128.1	119.0	120.8	23.37
7	0	1	1	0	0	1	1	0	127.5	129.5	126.0	123.6	124.9	122.1	7.21
8	1	1	1	0	1	0	0	0	124.0	127.4	129.1	125.2	116.5	117.9	25.87
9	0	0	0	1	0	1	1	1	122.6	126.8	123.3	122.8	129.7	122.9	8.62
10	1	0	0	1	1	0	0	1	121.2	127.7	124.1	128.4	124.3	125.6	6.98
11	0	1	0	1	1	0	1	0	125.2	125.6	121.8	123.4	122.6	121.4	3.13
12	1	1	0	1	0	1	0	0	124.7	127.6	133.3	124.5	119.1	123.0	22.72
13	0	0	1	1	1	1	0	0	128.0	128.1	125.3	121.6	117.8	120.2	18.41
14	1	0	1	1	0	0	1	0	116.5	132.5	129.5	124.4	120.3	118.9	39.86
15	0	1	1	1	0	0	0	1	129.4	132.6	122.3	126.7	128.0	128.8	11.68
16	1	1	1	1	1	1	1	1	120.8	127.5	124.8	124.2	128.2	118.6	14.04

Table 1.6: The Design Array and Data.

The  $\hat{D}_f$  estimates are in Table 1.7. and the  $\hat{D}_{f,g}$  estimates are in Table 1.8.

Factor	$f$	$\hat{D}_f$ or $\hat{D}_f^{-1}$	
A	1	$\hat{D}_1$	= 2.03
B	2	$\hat{D}_2^{-1}$	= 1.35
C	3	$\hat{D}_3$	= 1.81
D	4	$\hat{D}_4^{-1}$	= 1.07
E	5	$\hat{D}_5^{-1}$	= 1.11
F	6	$\hat{D}_6^{-1}$	= 1.12
G	7	$\hat{D}_7$	= 1.18
H	8	$\hat{D}_8^{-1}$	= 1.21

Table 1.7:  $\hat{D}_f$  and  $\hat{D}_f^{-1}$  Estimates.

The larger value is reported for each factor pairing.

Factors	$f$	$g$	$\hat{D}_{f,g}$ or $\hat{D}_{f,g}^{-1}$	Factors	$f$	$g$	$\hat{D}_{f,g}$ or $\hat{D}_{f,g}^{-1}$
A B	1	2	$\hat{D}_{1,2} = 1.16$	C E	3	5	$\hat{D}_{3,5} = 1.16$
A C	1	3	$\hat{D}_{1,3}^{-1} = 1.07$	C F	3	6	$\hat{D}_{3,6}^{-1} = 1.70$
A D	1	4	$\hat{D}_{1,4}^{-1} = 1.03$	C G	3	7	$\hat{D}_{3,7}^{-1} = 1.03$
A E	1	5	$\hat{D}_{1,5}^{-1} = 1.61$	C H	3	8	$\hat{D}_{3,8}^{-1} = 1.05$
A F	1	6	$\hat{D}_{1,6}^{-1} = 1.05$	D E	4	5	$\hat{D}_{4,5}^{-1} = 1.70$
A G	1	7	$\hat{D}_{1,7} = 1.08$	D F	4	6	$\hat{D}_{4,6} = 1.16$
A H	1	8	$\hat{D}_{1,8}^{-1} = 1.70$	D G	4	7	$\hat{D}_{4,7}^{-1} = 1.07$
B C	2	3	$\hat{D}_{2,3}^{-1} = 1.61$	D H	4	8	$\hat{D}_{4,8}^{-1} = 1.61$
B D	2	4	$\hat{D}_{2,4}^{-1} = 1.05$	E F	5	6	$\hat{D}_{5,6} = 1.08$
B E	2	5	$\hat{D}_{2,5}^{-1} = 1.07$	E G	5	7	$\hat{D}_{5,7}^{-1} = 1.05$
B F	2	6	$\hat{D}_{2,6}^{-1} = 1.03$	E H	5	8	$\hat{D}_{5,8}^{-1} = 1.03$
B G	2	7	$\hat{D}_{2,7}^{-1} = 1.70$	F G	6	7	$\hat{D}_{6,7}^{-1} = 1.61$
B H	2	8	$\hat{D}_{2,8} = 1.08$	F H	6	8	$\hat{D}_{6,8}^{-1} = 1.07$
C D	3	4	$\hat{D}_{3,4} = 1.08$	G G	7	8	$\hat{D}_{7,8} = 1.16$

Table 1.8:  $\hat{D}_{f,g}$  and  $\hat{D}_{f,g}^{-1}$  Estimates.

The larger value is reported for each factor pairing.

Let  $\alpha$  be the level of significance for the testing and take  $\alpha = 0.05$ . Since this design is balanced with 16 design points and six replicates at each design point then all the critical values are  $\Delta = F_{(0.95,40,40)} = 1.69$ . From Table 1.7  $\hat{D}_1 = 2.03$  is greater than one and greater than  $\Delta$  so the decision is “place one equals 0”.

$\hat{D}_3=1.81$  so “place three is a 0”. No other decisions can be made from this table. From Table 1.8  $\hat{D}_{1,8}^{-1} = \hat{D}_{2,7}^{-1} = \hat{D}_{3,6}^{-1} = \hat{D}_{4,5}^{-1} = 1.70$  which is just greater than  $\Delta$ . All of these interactions are aliases of each other so that explains why their estimates are equal and their  $S^2(u, v)$  values are all the same. The  $S^2(u, v)$ 's are in in Table 1.9.

$S^2(u, v)$	$v = 0$	$v = 1$
$u = 0$	8.08	13.31
$u = 1$	27.42	15.95

Table 1.9: The  $S^2(u, v)$  values for factors A and H, B and G, C and F and D and E.

$\text{Min}[ S_{f,g}^2(0, 0), S_{f,g}^2(1, 1) ] = S_{f,g}^2(0, 0) = 8.08$  and

$\text{Min}[ S_{f,g}^2(0, 1), S_{f,g}^2(1, 0) ] = S_{f,g}^2(0, 1) = 13.31$ . The former is smaller so the decisions are

“place 1 equals place 8”, “place 2 equals place 7”,  
 “place 3 equals place 6”, “place 4 equals place 5”.

The set of agreements is

$$\begin{aligned} & \{(\text{place } 1=0, \quad \text{place } 3=0, \\ & \text{place } 1=\text{place } 8, \quad \text{place } 3=\text{place } 6)\} \\ & = \{(0, X, 0, X, X, 0, X, 0)\} \end{aligned}$$

where  $X$  can equal 0 or 1.

The set of disagreements is  $\{(\text{place } 2=\text{place } 7), (\text{place } 4=\text{place } 5)\}$ . These are split into sets in which only those within each set disagree with each other.

$$G1 = \{(\text{place } 2 = \text{place } 7)\} = \{(X, 0, X, X, X, X, 0, X), (X, 1, X, X, X, X, 1, X)\}$$

and

$$G2 = \{(\text{place } 4 = \text{place } 5)\} = \{(X, X, X, 0, 0, X, X, X), (X, X, X, 1, 1, X, X, X)\}$$

The maximal subsets of  $G1$  are  $\{(X, 0, X, X, X, 0, X)\}$  and  $\{(X, 1, X, X, X, X, 1, X)\}$  and for  $G2$   $\{(X, X, X, 0, 0, X, X, X)\}$  and  $\{(X, X, X, 1, 1, X, X, X)\}$ .

Finally each set in the maximal disagreement sets is intersected with the agreement set. For example

$$(0, X, 0, X, X, 0, X, 0) \cap (X, 0, X, X, X, 0, X) \cap (X, X, X, 0, 0, X, X, X)$$



$$= (0, 0, 0, 0, 0, 0, 0, 0)$$

The complete set of runs is

$$\{(0, 0, 0, 0, 0, 0, 0, 0), (0, 0, 0, 1, 1, 0, 0, 0), (0, 1, 0, 0, 0, 0, 0, 1), (0, 1, 0, 1, 1, 0, 0, 1)\}$$

Since the factors A (place 1), C (place 3), F (place 6), and H (place 8) are fixed, in this case at zero, this suggests that they are variance factors that affect the variance of the process. From the construction of the data it is known that only A and C are factors that affect the variance of the process. However for  $\alpha=0.0495$  the correct decisions are made. This method requires sensitivity analysis on the value of  $\alpha$  as the outcomes can change as  $\alpha$  changes.

If the response is squared then the response is no longer Normally distributed. The method gives  $\{(0, 0, 0, 0, 0, 0, 0, 0)\}$  as the outcome. This signifies that all the factors are variance control factors which is not the case.

### Comments

This method allows for the implicit introduction of the noise factors.

The assumption that each point in the design space is unbiased to the Target value is too restrictive. It may be possible to manipulate this through the use of mean control factors that are not variance control factors. However the point of this method is to identify the factors that are variance control factors; having this knowledge prior to the experiment begs the question of why the experiment should be done at all.

The notation for this method is very unwieldy which can be offputting although it represents very simple concepts. The method taken to get to a solution feels foreign in this area of Quality Control e.g. taking unions and intersections of sets. It is not easy to relate what is happening in the method back to the system under study.

The method is not robust to non-Normal distributions however the use of the F test can be justified by the Central Limit Theorem as long as the experiment has a reasonably large number of design points and/or replicates. For few replicates the estimates of  $\sigma^2$  will not be good.

The choice of  $\alpha$  has also to be carefully considered as a large number of hypothesis tests are done. The outcomes can vary depending on the choice of  $\alpha$

therefore sensitivity analysis should be done to see how the solution set of runs changes as  $\alpha$  changes.

Careful consideration of alias effects has to be taken after the decisions have been made as these can affect the outcome of the places in the run vector. If there are no subjective reasons why one interaction or effect in an alias string is preferred then taking into account all possible aliased effects as being possible effects will make the solution set large.

The output allows for multiple solutions but gives no hints as to which one of those is preferred. If there are a large number of runs in the optimal set, it is not easy to work out what the nature of the effects are. This makes it difficult to decide what direction to take for further investigation.

### 1.9.2 Transformations and a Data Analytic Approach

One of the problems of the Taguchi technique for Parameter Design was that the  $s/n$  ratios can be affected by large mean effects. One example of this is when the mean is proportional to the variance. Box [7] proposed a method which is based on transforming the data to remove the relationship between the mean and variance.

After performing an experiment the relationship between the mean and variance should be analysed and once the relationship had been discovered the original data should be transformed to remove it. An ANOVA is then done on the means of the transformed data to work out the factors that can be used to adjust the level of a process to a target. Similarly an ANOVA is done on the log standard deviations of the transformed data to work out those factors that affect the variance of the process so they can be used to minimise the variance.

Box proposed the Box-Cox transformations as the a set of appropriate transformations to use. They are defined as

$$x^{(\lambda)} = \begin{cases} \frac{x^\lambda - 1}{\lambda} & \lambda \neq 0 \\ \ln x & \lambda = 0 \end{cases}$$

and are continuous in  $\lambda$  for  $x > 0$ .

The transformations are used to achieve two characteristics - parsimony, where the model is as uncomplicated as possible and separation, where the variance and mean are not related to each other.

Box showed how a lambda plot could be used to find the correct transformation and Nair and Pregibon also gave some further graphical methods for analysing the data both in [7]. This is a combination of both their methods.

- Start with a box plot for each experimental run. Some pattern in the data may give an indication of which factors affect the mean and the variance. Also any outliers can be observed. Outliers are a problem since they inflate the variance. Careful consideration has to be given as to whether it is random chance that has caused them to appear, some kind of error or a true reflection of the system.
- Another exploratory tool is a Mean-Variance plot. The mean and variance of each experimental run are plotted against each other on log scales. This will show up any relationship of the type  $\sigma^2 \propto \mu^\alpha$ . A least squares slope is drawn. No dependence is observed if the data appears to be random and the slope of the line appears to be insignificant.
- Lambda plots for both the mean effects and the log standard deviation effects are drawn. This plots the t-value of an effect against the value of lambda that has been used to transform the data. The points for each factor are joined and the value of  $\lambda$  where the absolute value of the derivative of the slope for each factor is the smallest is considered. Hopefully the values found for  $\lambda$  will be very similar for those factors deemed significant. To achieve parsimony,  $\lambda$  is observed at the place on the plot where there is the least complication, where as many of the points lie as close to  $t=0$  as possible. The points that are significantly different from 0 are the ones that will be included in the model.

The t-values for the mean effects can be worked out from the within-run sums-of-squares. The way I preferred to do this however, was to fit the model to all the data, instead of just the experimental means, by computer, which still gives the correct values for the effects of the means and the t-values for the effects as well. The value for the variance of the log standard deviation effects are found by using  $\sigma_{\log s} \approx \frac{1}{2(n-1)}$ .

- Once a value for lambda has been decided upon, QQPlots for the mean and log standard deviation effects are drawn. The slope of the line through the

non-significant factors in both graphs are recorded as a measure of the error variance.

- Finally a cross plot is drawn of the mean effects,  $x$ , versus the log standard deviation effects,  $y$ , and filter lines, are drawn in. The filter lines are drawn in at  $x = \pm 2\sigma_l$  and  $y = \pm 2\sigma_d$  where  $\sigma_l$  is the slope of the line through the non-significant mean effects and  $\sigma_d$  is the slope of the line through the non-significant log standard deviation effects. Where the factors fall in this graph gives an indication of what type of factor it is. There are four zones in the graph;
  - zone 1. The centre box of the cross arms is where all the non-significant factors lie.
  - zone 2. The horizontal cross arms, outside the center box, are where the significant mean effects lie.
  - zone 3. The vertical cross arms, outside the center box, are where the significant log standard deviation effects lie.
  - zone 4. The areas outside the crossarms are where both significant mean and log standard effects lie.

This is displayed in Figure 1.9.

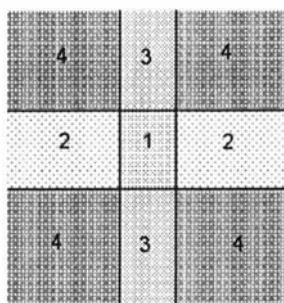


Figure 1.9: The Special Areas of a Cross Plot

### Example

This Section gives an example of this method at work. A  $F 2_4^{8-4}$  design was used to construct an experiment. The data was normally distributed with factor A and

D being mean effects and factors A and C being variance effects. The overall mean was 125, factor A=10 and D = 15. The variances were

A	C	Variance
-1	-1	4
1	-1	11
-1	1	12
1	1	33

where the  $\pm 1$ 's indicate the level of the factors. Six replicates were taken at each design point. The resulting data was divided by ten and squared to give the working data outlined in Table 1.10. Note the data has been truncated to integers.

Table 1.10:  $F2_4^{8-4}$  Design plus the Response Data

A	B	C	D	E	F	G	H	$y_1$	$y_2$	$y_3$	$y_4$	$y_5$	$y_6$
-1	-1	-1	-1	-1	-1	-1	-1	100	101	94	98	105	96
1	-1	-1	-1	1	1	1	-1	150	140	129	137	149	123
-1	1	-1	-1	1	1	-1	1	94	101	105	103	106	109
1	1	-1	-1	-1	-1	1	1	143	150	129	151	138	158
-1	-1	1	-1	1	-1	1	1	103	103	89	85	112	93
1	-1	1	-1	-1	1	-1	1	153	145	126	151	129	134
-1	1	1	-1	-1	1	1	-1	105	109	102	97	99	94
1	1	1	-1	1	-1	-1	-1	141	149	153	144	124	127
-1	-1	-1	1	-1	1	1	1	162	173	164	163	181	163
1	-1	-1	1	1	-1	-1	1	213	233	222	235	222	226
-1	1	-1	1	1	-1	1	-1	169	170	160	164	162	159
1	1	-1	1	-1	1	-1	-1	224	232	250	223	207	218
-1	-1	1	1	1	1	-1	-1	176	177	169	160	150	156
1	-1	1	1	-1	-1	1	-1	200	248	238	223	211	206
-1	1	1	1	-1	-1	-1	1	180	189	161	173	176	178
1	1	1	1	1	1	1	1	212	232	224	222	234	206

The generators for this design are  $I=ABCE$ ,  $I=ABDF$ ,  $I=ACDG$  and  $I=BCDH$ . They are all four characters long confirming that this is a resolution 4 design. The aliases are listed below. The number associated with them will be used in the plots to follow;

1. AF, BD, CH, EG
2. AH, BG, DE, CF
3. AC, DG, HF, BE
4. AB, DF, CE, HG
5. AD, EH, CG, BF
6. AE, BC, FG, DH
7. AG, BD, EG, CH.

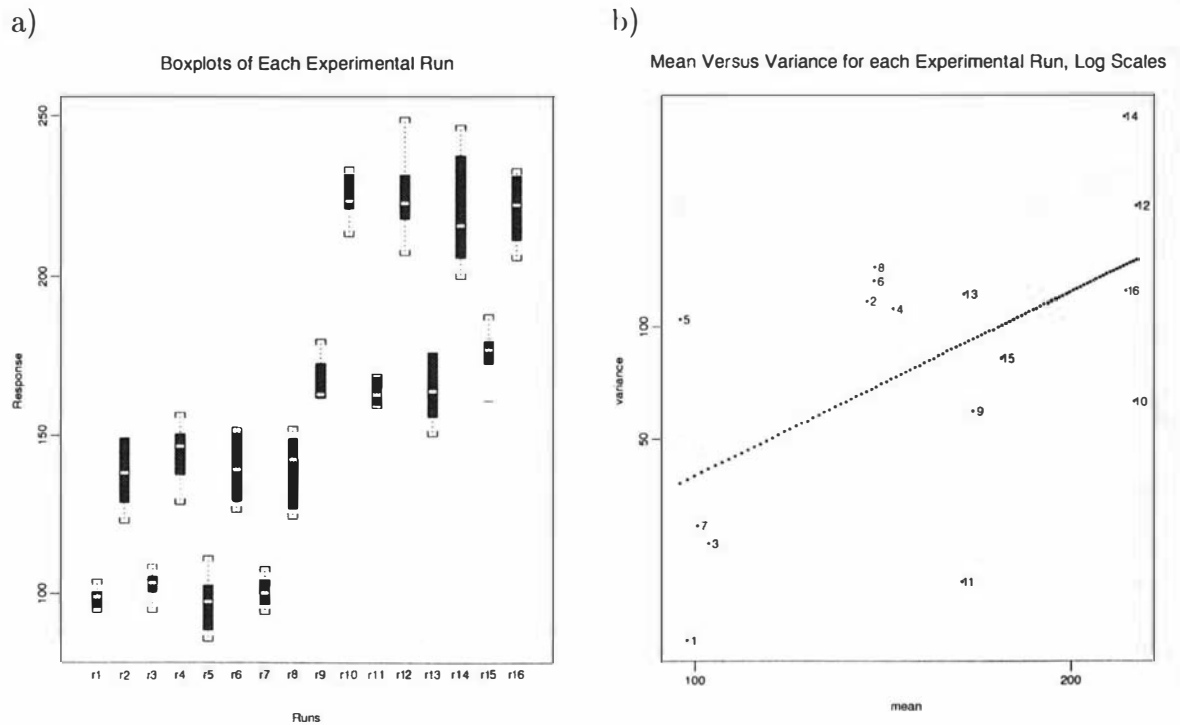


Figure 1.10: a) Boxplot and b) Mean-Variance Plot of the Response Data

In Figure 1.10 a) the boxplot of the data is shown. After observing the design matrix as well it is apparent that factors A and D are mean control factors i.e. first group of boxplots are centred about a different value than the second group and alternating boxplots are centred about different values. Although there appears to be differences in the variances for the runs there does not seem to be

any pattern to it. The Figure 1.10 b) gives an indication that the mean and variance are interlinked, therefore a Taguchi type analysis of this data may not give the correct sets of control factors. The value of the slope is  $\hat{\beta}_1 = 1.64$  has  $p$ -value=0.0237 indicating that the dependence between the mean and variance is significant. The relationship is  $\log s^2 = 1.64 \log \bar{x} - 3.89$  or  $s^2 \propto \bar{x}^{1.64}$

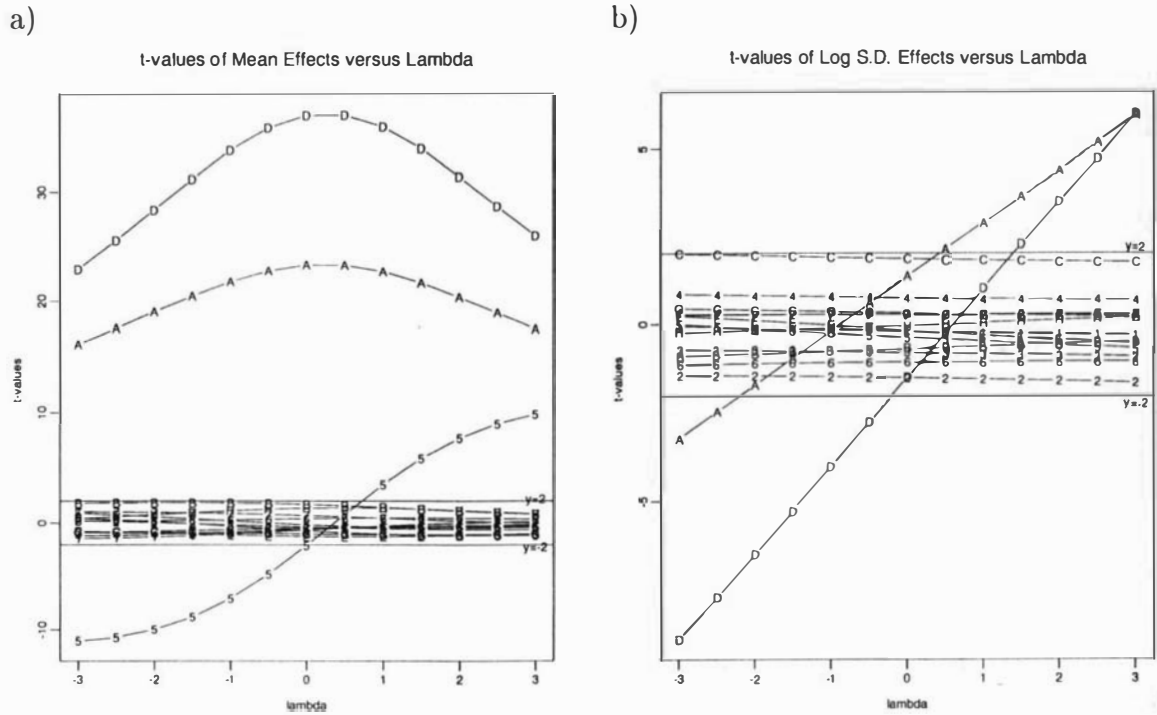


Figure 1.11: Lambda Plots of a) Mean Effects and b) Log S.D. Effects after transformation

The lambda plot of the mean effects is shown in Figure 1.11 a). What we are looking for in this plot is a value for lambda where the plot is simplest and the change in slope of the significant factors is smallest in absolute value. Lines have been drawn at  $t = \pm 2$  to give an indication of the non-significant factors. The lines labelled A and D have smallest gradient in absolute value at approximately  $\lambda \in [0, 1]$ . Line 5 is the AD interaction (although without prior knowledge it could also be the aliases of AD i.e.  $EH=CG=BF$ ). It passes through zero at  $\lambda = 0.5$ . Since a model with no interactions is simpler than a model with interactions this value would give the model the characteristic of parsimony.

The lambda plot of the log s.d. effects is shown in Figure 1.11 b). In this plot the only significant effects are A and D although C may also be. If we look at parsimony then two possible values for lambda are  $\lambda = -1$  where the t-value for factor A is zero or  $\lambda = 0.5$  where the t-value for factor D is zero. The gradients of all slopes are unchanging. In this plot there are two possible choices for lambda but only one that agrees with that from the first plot therefore we would choose 0.5 for the value for  $\lambda$ .

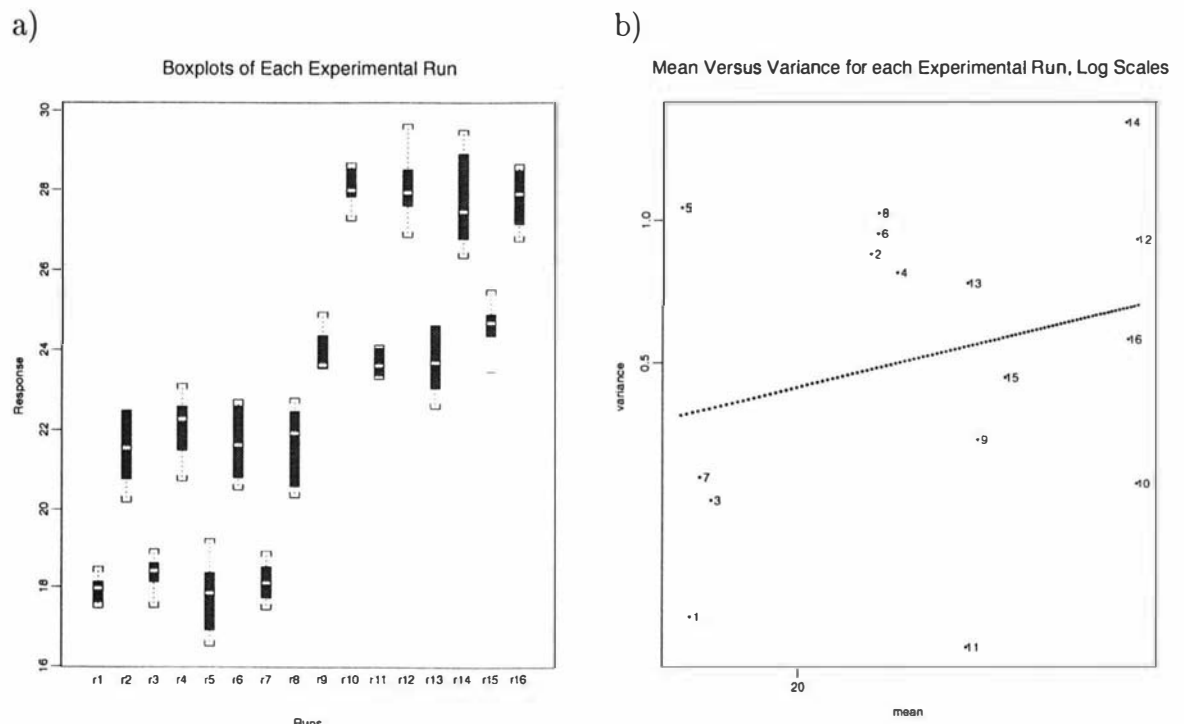


Figure 1.12: a) Boxplot and b) Mean-Variance Plot of the Transformed Response Data

The data has been transformed by a Box-Cox transformation with  $\lambda = 0.5$  and the boxplot in Figure 1.12 a) does not appear to add much information to that which is already known. The Mean-Variance plot (Figure 1.12 b) shows a more random scattering of the data and although there is a non-zero slope it is non-significant with a p-value of 0.3436. This confirms that we have found a suitable metric in which to analyse the dispersion effects without the mean effects clouding the issue.



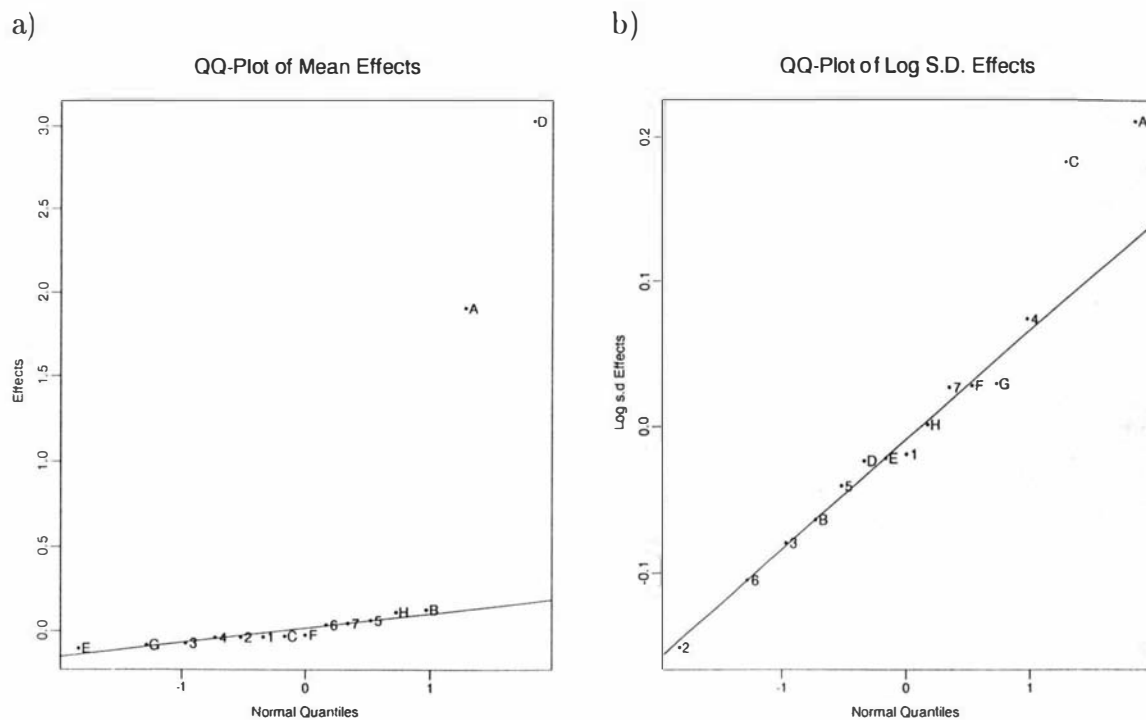


Figure 1.13: QQPlots of a) Mean Effects and b) Log S.D. Effects after transformation

The QQPlot of the mean effects (Figure 1.13 a) with a least squares line through the estimated non-significant effects shows clearly that factors A and D are significant mean effects. The QQPlot of the log standard deviation effects (figure 1.13 b) with a least squares line also through the estimated non-significant effects shows that factors A and C are significant dispersion effects.

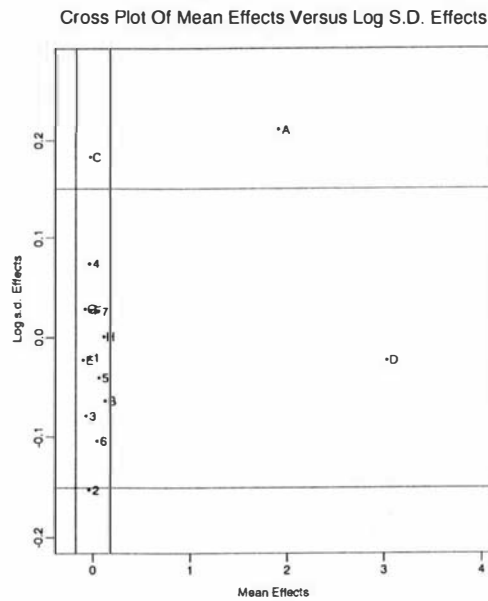


Figure 1.14: Cross Plot of Transformed Data

The final figure is the crossplot (Figure 1.14). Only the factors A, C, D and interaction 2 (the AH=BG=DE=CF interaction) lie outside zone 1. Interaction 2, since it lies in zone 3, affects the variance of the process. In the QQPlot interaction 2 had a large effect but was consistent with being on the line of insignificant effects therefore interaction 2 will be presumed to fall (just) outside zone 1 by chance alone. Factor C is in zone 3 indicating it is a variance control factor. Factor D is in zone 2 indicating it is a mean control factor and factor A is in zone 4 indicating it is both a mean and variance control factor. Reading off the axes or doing an ANOVA gives the value of the effects. Noting that  $y$  is the response variable, the model for this data is therefore

$$y^* = \frac{y^{0.5} - 1}{0.5},$$

$$\mu_{y^*} = 22.91 + 1.90A + 3.03D$$

and

$$\log(s_{y^*}) = 0.33 + 0.21A + 0.18C.$$

Since all the coefficients in the variance model are positive the variance will be least when factors A and C are set at their low levels. A is fixed in the mean model by the minimisation of the variance model and the level of the process can

be changed through factor D. It may be that manipulating factor D will not bring the process to the level required and A may have to be changed to achieve that level with some increase in variance.

### Comments

This method allows for the uncontrollable factors to be implicitly set.

The positive feature of this method is the use it makes of graphical methods. The graphs allow for a good understanding of the modelling processes involved and the relationships between the factors.

Instances occur when a single transformation will not bring about the properties of parsimony and separation. Removing the relationship between the mean and variance when modelling the variance is preferable i.e. separation. The model also assumes that the transformation will cause the response variable to be normally distributed. This may not happen either. Of course if the amount of data is large the Central Limit Theorem will hold and the effects will be Normally Distributed and the QQPlot becomes valid.

With fractionated designs of resolution three discerning between a main effect or its aliased interaction is impossible, and using lambda plots to get a parsimonious model maybe very difficult. With very few factors, especially if many are significant, it may be impossible to judge from the Lambda Plot what the correct transformation could be.

From experience, estimation of the variances requires many more replicates for each design point when compared with the mean estimates. This means this method can be quite expensive because of the size required to get good estimates.

### 1.9.3 Response Surface Methods

Myers et al[24] used the ideas of RSM to simultaneously give a surface for the expected response and variance of the response. The first surface is used to find values of the factors that either achieve a target, a minimum or maximum depending on the intent of the experiment. The second surface is used to find values of the factors that give minimum variance. On many occasions these two sets of values may not coincide and a subjective analysis of which objective is the more important will have to be done.

### The Method

In this method there are two sets of factors, control factors and uncontrollable (noise) factors. The noise factors produce the heterogeneous variance in the process while the control factors do not.

A RSM design is used where the control factors,  $x_j$ ,  $j = 1, \dots, p$ , and noise factors,  $z_k$ ,  $k = 1, \dots, q$ , are treated identically. The  $i^{\text{th}}$  run is made up of  $x_{i,1}, \dots, x_{i,p}, z_{i,1}, \dots, z_{i,q}, y_i$  where  $y_i$  is the response.

The model that is fitted is

$$y_i = \beta_0 + \mathbf{g}'(\mathbf{x}_i)\boldsymbol{\beta} + \mathbf{z}'_i\boldsymbol{\delta} + \mathbf{g}'(\mathbf{x}_i)\boldsymbol{\Lambda}\mathbf{z}_i + \epsilon_i.$$

where  $\boldsymbol{\beta}$  and  $\boldsymbol{\delta}$  are vectors of coefficients.  $\mathbf{g}'(\mathbf{x}_i)$  is a row from the matrix of independent variables,  $\mathbf{X}$ , with the first number removed. This number would be a one which is used in the estimation of  $\beta_0$  but here  $\beta_0$  is estimated separately.  $\boldsymbol{\Lambda}$  is a matrix of interaction parameters between the  $x_j$  and  $z_k$ . The  $x_j$  and  $z_k$  are chosen at fixed levels.

Although the  $z_k$ 's have been taken to be fixed in the first surface they are in fact random variables and this is exploited in finding the variance response surface.

The process variance is

$$\begin{aligned} \text{var}(y(x)) &= \text{var}([\boldsymbol{\delta}' + \mathbf{g}'(\mathbf{x})\boldsymbol{\Lambda}]\mathbf{z} + \epsilon) \\ &= [\boldsymbol{\delta}' + \mathbf{g}'(\mathbf{x})\boldsymbol{\Lambda}]\mathbf{V}[\boldsymbol{\delta}' + \mathbf{g}'(\mathbf{x})\boldsymbol{\Lambda}]' + \sigma_\epsilon^2 \end{aligned}$$

where  $\mathbf{V}$  is the variance-covariance matrix of  $\mathbf{z}$ .  $\mathbf{V}$  is not estimated in the experiment and so it must be estimated or guessed from other data.

The  $z_k$ 's are centered around zero and hence their expectation is zero. The estimated response surface for the mean is therefore

$$\hat{y}(x) = \hat{\beta}_0 + \mathbf{g}'(\mathbf{x})\hat{\boldsymbol{\beta}}$$

and the variance response surface is

$$\widehat{\text{var}}(\hat{y}(x)) = [\hat{\boldsymbol{\delta}}' + \mathbf{g}'(\mathbf{x})\hat{\boldsymbol{\Lambda}}]\mathbf{V}[\hat{\boldsymbol{\delta}}' + \mathbf{g}'(\mathbf{x})\hat{\boldsymbol{\Lambda}}]' + \hat{\sigma}_\epsilon^2.$$

This model assumes that there are no interactions between the  $z_k$ 's.

**Example**

A RSM experiment was run with the design, a  $F2^4$  with four center points, and data given in Table 1.11. There were two control factors and two noise factors labeled  $x_1$ ,  $x_2$ ,  $z_1$  and  $z_2$  respectively.

Design				Data
$x_1$	$x_2$	$z_1$	$z_2$	$y$
0	0	0	0	97.70483
0	0	0	0	96.47317
0	0	0	0	100.72425
0	0	0	0	101.17510
-1	-1	-1	-1	95.22444
-1	-1	-1	1	104.42875
-1	-1	1	-1	87.67168
-1	-1	1	1	95.66690
-1	1	-1	-1	99.87573
-1	1	-1	1	95.88072
-1	1	1	-1	89.58390
-1	1	1	1	80.90409
1	-1	-1	-1	75.24423
1	-1	-1	1	82.74909
1	-1	1	-1	93.53634
1	-1	1	1	101.74036
1	1	-1	-1	118.17977
1	1	-1	1	115.46116
1	1	1	-1	136.44676
1	1	1	1	129.08517

Table 1.11: RSM Design and Data.

The estimated parameters for the model are given in Table 1.12.

	<i>coef</i>	<i>std.err</i>	<i>t.stat</i>	<i>p.value</i>
<i>Intercept</i>	99.8878	0.3850	259.4163	0.0000
$x_1$	6.4504	0.4305	14.9836	0.0000
$x_2$	8.0722	0.4305	18.7509	0.0000
$z_1$	1.7245	0.4305	4.0057	0.0031
$z_2$	0.6346	0.4305	1.4741	0.1746
$x_1x_2$	10.1656	0.4305	23.6137	0.0000
$x_1z_1$	6.9223	0.4305	16.0799	0.0000
$x_1z_2$	0.0690	0.4305	0.1603	0.8762
$x_2z_1$	-0.8966	0.4305	-2.0828	0.0670
$x_2z_2$	-3.4790	0.4305	-8.0813	0.0000
$z_1z_2$	-0.6149	0.4305	-1.4283	0.1870

Table 1.12: Table of Coefficients for RSM Design.

The terms  $z_2$ ,  $x_1z_2$ ,  $x_2z_1$  and  $z_1z_2$  were found not to be significant at the 5% level and were discarded from the model. An ANOVA was done with the remaining terms and is presented in Table 1.13. The variance of the center points was used for the pure error mean square and the residual error was used for the total error to work out the lack of fit. At the 5% there was no evidence of lack of fit as  $\Pr(F)=0.29$ .

Analysis of Variance Table					
	<i>Df</i>	<i>Sum of Sq</i>	<i>Mean Sq</i>	<i>F Value</i>	<i>Pr(F)</i>
$x_1$	1	665.73	665.73	166.05	0.00
$x_2$	1	1042.57	1042.57	260.05	0.00
$z_1$	1	47.58	47.58	11.87	0.00
$x_1x_2$	1	1653.44	1653.44	412.42	0.00
$x_1z_1$	1	766.70	766.70	191.24	0.00
$x_2z_2$	1	193.65	193.65	48.30	0.00
<i>Residuals</i>	13	52.12	4.01		

	<i>Df</i>	<i>Sum of Sq</i>	<i>Mean Sq</i>	<i>F Value</i>	<i>Pr(F)</i>
<i>Lack of Fit</i>	10	36.35	3.64	0.69	0.29
<i>Pure Error</i>	3	15.76	5.25		
<i>Total Error</i>	13	52.12	4.01		

Table 1.13: ANOVA for RSM Design.

The model was taken as

$$\hat{y} = 99.9 + 6.5x_1 + 8.1x_2 + 1.7z_1 + 10.2x_1x_2 + 6.9x_1z_1 - 3.5x_2z_2.$$

in terms of the notation this is

$$\begin{aligned} \hat{\beta}_0 = 99.9, \quad \mathbf{g}(\mathbf{x}) &= \begin{pmatrix} x_1 \\ x_2 \\ x_1x_2 \end{pmatrix}, \quad \hat{\boldsymbol{\beta}} = \begin{pmatrix} 6.5 \\ 8.1 \\ 10.2 \end{pmatrix}, \\ \mathbf{z} = \begin{pmatrix} z_1 \\ z_2 \end{pmatrix}, \quad \hat{\boldsymbol{\delta}} = \begin{pmatrix} 1.7 \\ 0 \end{pmatrix}, \quad \hat{\boldsymbol{\Lambda}} = \begin{pmatrix} 6.9 & 0 \\ 0 & -3.5 \\ 0 & 0 \end{pmatrix}, \\ \hat{\sigma}_\epsilon^2 &= 5.3. \end{aligned}$$

The response surface for the mean is

$$\begin{aligned} \hat{y}(x) &= \hat{\beta}_0 + \mathbf{g}'(\mathbf{x})\hat{\boldsymbol{\beta}} \\ &= 99.9 + 6.5x_1 + 8.1x_2 + 10.2x_1x_2 \end{aligned} \tag{1.10}$$

and for the variance response surface

$$\widehat{\text{var}}(\hat{y}(x)) = [\hat{\boldsymbol{\delta}}' + \mathbf{g}'(\mathbf{x})\hat{\boldsymbol{\Lambda}}]\mathbf{V}[\hat{\boldsymbol{\delta}}' + \mathbf{g}'(\mathbf{x})\hat{\boldsymbol{\Lambda}}]' + \hat{\sigma}_\epsilon^2$$

$$\begin{aligned}
&= \left[ (1.7, 0) + (x_1, x_2, x_1x_2) \begin{pmatrix} 6.9 & 0 \\ 0 & -3.5 \\ 0 & 0 \end{pmatrix} \right] \mathbf{V} \times \\
&\quad \left[ (1.7, 0) + (x_1, x_2, x_1x_2) \begin{pmatrix} 6.9 & 0 \\ 0 & -3.5 \\ 0 & 0 \end{pmatrix} \right]' + 5.25 \\
&= [(1.7, 0) + (6.9x_1, -3.5x_2)] \mathbf{V} [(1.7, 0) + (6.9x_1, -3.5x_2)]' + 5.3
\end{aligned}$$

At this point either the variance-covariance matrix of  $\mathbf{z}$  has to be guessed or estimated by sampling. Suppose that  $\mathbf{V}$  has been estimated as

$$\mathbf{V} = \begin{pmatrix} 2 & 0 \\ 0 & 4 \end{pmatrix}$$

This gives for  $\widehat{\text{var}}(\hat{y}(x))$

$$\begin{aligned}
\widehat{\text{var}}(\hat{y}(x)) &= (1.7 + 6.9x_1, -3.5x_2) \begin{pmatrix} 2 & 0 \\ 0 & 4 \end{pmatrix} \begin{pmatrix} 1.7 + 6.9x_1 \\ -3.5x_2 \end{pmatrix} + 5.25 \\
&= 2(1.7 + 6.9x_1)^2 + 4(-3.5x_2)^2 + 5.3 && (1.11) \\
&= 18.1 + 47.8x_1 + 47.9x_1^2 + 193.7x_2^2
\end{aligned}$$

From equations 1.10 and 1.11 the surfaces for the mean and variance surfaces can be drawn. The response surface for the mean is in Figure 1.15.



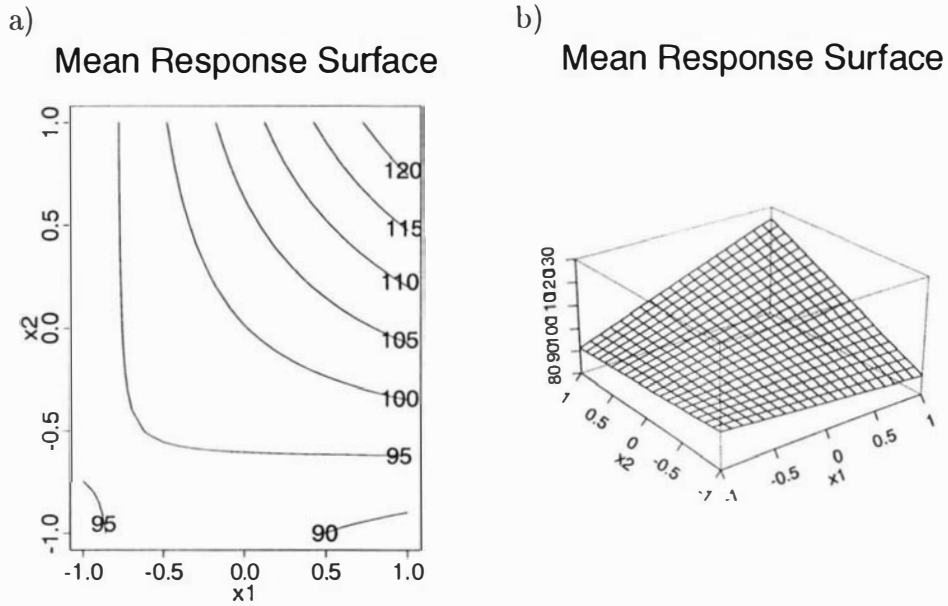


Figure 1.15: a) Contour Plot and b) Perspective Plot of the Mean Response Surface.

The response surface for the variance is in Figure 1.16.

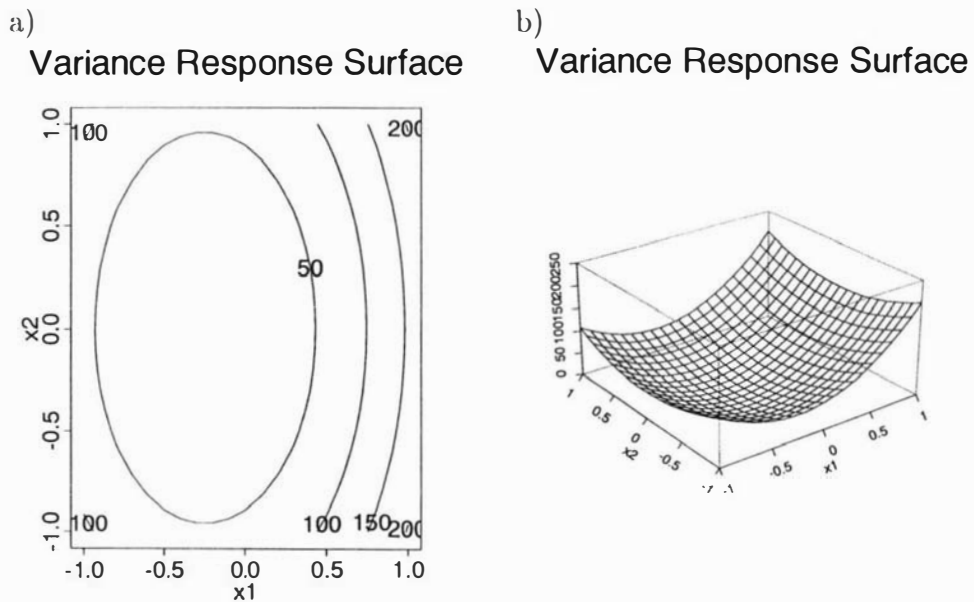


Figure 1.16: a) Contour Plot and b) Perspective Plot of the Variance Response Surface.

Using equation (1.11) the point of minimum variance is  $x_1 = -1.7/6.9 = -0.25$

and  $x_2 = 0$  and the variance at that point is 5.3. At that point on the mean response surface the value for the mean is 98.3 found using equation (1.10). Suppose that the target value was  $y = 100$  then we need to operate the process at  $x_1$  and  $x_2$  such that

$$100 = 99.9 + 6.5x_1 + 8.1x_2 + 10.2x_1x_2 \quad (1.12)$$

and

$$\widehat{\text{var}}(\hat{y}(x)) = 2(1.7 + 6.9x_1)^2 + 4(-3.5x_2)^2 + 5.3 \quad (1.13)$$

is a minimum. Solving equation (1.12) for  $x_2$  and substituting this in equation (1.13) and solving numerically for  $x_1$ , then  $x_2$ , gives  $x_1 = -0.15$ ,  $x_2 = 0.16$  with the variance at this point being 7.5. This is shown graphically in Figure 1.17.

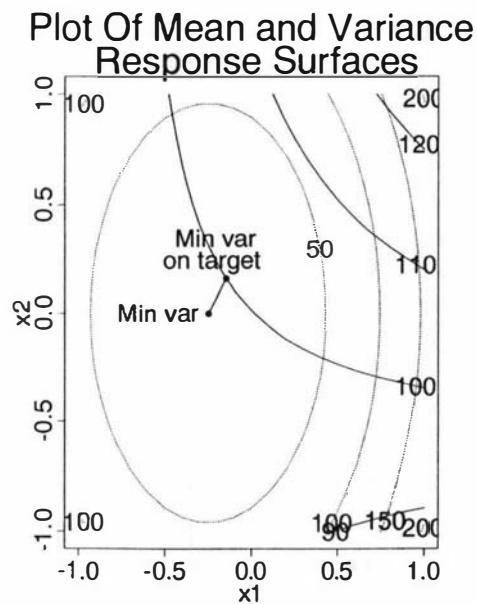


Figure 1.17: Joint Contour Plot of the Mean Response Surface and Variance Response Surface.

A balance has to be made between minimising the variance and keeping the mean on target. Only one of these may be achieved at the expense of the other.

### Comments

The uncontrollable factors are explicitly set in the experimental designs for this method.

The major criticism of this method is that interactions between the  $z_k$ 's are not catered for. A preferable model would be

$$y_i = \beta_0 + \mathbf{g}'(\mathbf{x}_i)\boldsymbol{\beta} + \mathbf{z}'_i\boldsymbol{\delta} + \mathbf{g}'(\mathbf{x}_i)\boldsymbol{\Lambda}\mathbf{z}_i + \mathbf{z}'_i\boldsymbol{\Upsilon}\mathbf{z}_i + \epsilon_i.$$

where  $\boldsymbol{\Upsilon}$  is a matrix of interaction parameters between  $z_j$  and  $z_k$ . The reason for leaving that term out is that the variance of  $y_i$  for this model is extremely hard to find as it involves higher moments of the  $\mathbf{z}$ 's. In a very special case if  $\boldsymbol{\Upsilon}$  is a projection matrix and  $\mathbf{z}$  is a standard normal vector then  $\mathbf{z}'\boldsymbol{\Upsilon}\mathbf{z}$  is distributed as  $\chi_p^2$  where  $p$  is the rank of  $\boldsymbol{\Upsilon}$ .  $\boldsymbol{\Upsilon}$  is unlikely to be a projection matrix.

The other criticism of this method is that the noise and control variables have constraints on their behaviour i.e. noise variables produce the heterogeneous variance while the control variables do not. Sometimes the choice of what is an uncontrollable variable and what is not is dependent on the beliefs of the experimenter therefore having the distinction between the two is not practicable. This method was extended further by Engel and Heule [14] to include the case when there is a non-constant variance of the residuals by using generalised linear models.

#### 1.9.4 Split-Plot Methods.

Box and Jones [8] proposed the use of split plot experimental designs as an alternative method to Dr. Taguchi's method of Parameter Design. One of the main criticisms of Taguchi's method of Parameter Design is that the designs can become large when the inner and outer arrays are crossed. This means that the amount of experimental work is large which increases the complexity and cost. Split plot designs offer a way of arranging the sequence of runs in more convenient manner so that there is less experimental work. This arrangement of runs may reduce the complexity or the cost or both. In some cases the physical constraints within the system under study may mean that split plot designs may be the only design possible.

The idea of having uncontrollable and controllable factors in split plot designs to improve quality is not a recent development. An example of testing four washing products in four washing machines with the noise factors of temperature and water was reported by Michael [22] in 1964.

## Method

The extension from the classical use of split plot designs to the use of split plot designs for the analysis of mean and dispersion effects is not large. Here the uncontrollable factors are put in the whole plots and the controllable factors are put in the split plots.

Box and Jones [8] proposed that there be  $n$  levels of the controllable factors,  $C_1, \dots, C_i, \dots, C_n$ , applied to the split plots,  $m$  levels of the uncontrollable factors,  $U_1, \dots, U_j, \dots, U_m$ , applied to the whole plots and that there be  $l$  replicates,  $r_1, \dots, r_k, \dots, r_l$  with the whole plots in  $l$  randomized blocks.

The model for the arrangement is

$$y_{ijk} = \mu + r_k + U_j + \eta_{jk} + C_i + (UC)_{ij} + \epsilon_{ijk}$$

where  $y_{ijk}$  is the  $k^{\text{th}}$  replicate of the  $i^{\text{th}}$  level of the Controllable factor  $C$  and the  $j^{\text{th}}$  level of the Uncontrollable factor  $U$ ,  $\mu$  is the overall mean,  $r_k$  is the random effect of the  $k^{\text{th}}$  replicate with  $r_k$  being distributed as  $N(0, \sigma_r^2)$ ,  $U_j$  is the fixed effect of the  $j^{\text{th}}$  level of factor  $U$ ,  $C_i$  is the fixed effect of the  $i^{\text{th}}$  level of factor  $C$ ,  $(UC)_{ij}$  is the interaction of the  $i^{\text{th}}$  level of factor  $U$  with the  $j^{\text{th}}$  level of factor  $C$ ,  $\eta_{jk}$  is the whole plot error and  $\eta_{jk}$  is distributed as  $N(0, \sigma_w^2)$ . The subplot error,  $\epsilon_{ijk}$ , is distributed as  $N(0, \sigma_s^2)$ . The  $\epsilon_{ijk}$  are independent.

The ANOVA table is in Table 1.14 where  $\hat{U}_j$ ,  $\hat{C}_i$  and  $\widehat{UC}_{ij}$  are estimates of  $U_j$ ,  $C_i$  and  $UC_{i,j}$  respectively.

Source	d.f.	Sum of Squares	Expected Mean Square (MSE)
Whole Plot			
Reps( $R$ )	$l - 1$		$mn\sigma_r^2 + \sigma_s^2 + n\sigma_w^2$
$U$	$m - 1$	$nl \sum_{j=1}^m \hat{U}_j^2$	$\frac{nl}{m-1} \sum_{j=1}^m U_j^2 + \sigma_s^2 + n\sigma_w^2$
$R \times U$ (Error 1)	$(l-1)(m-1)$		$\sigma_s^2 + n\sigma_w^2$
Split Plot			
$C$	$n - 1$	$lm \sum_{i=1}^n \hat{C}_i^2$	$\frac{lm}{n-1} \sum_{i=1}^n C_i^2 + \sigma_s^2$
$U \times C$	$(n-1)(m-1)$	$l \sum_{i=1}^n \sum_{j=1}^m (\widehat{UC})_{ij}^2$	$\frac{l}{(n-1)(m-1)} \sum_{i=1}^n \sum_{j=1}^m (UC)_{ij}^2 + \sigma_s^2$
Error 2	$(l-1)m(n-1)$		$\sigma_s^2$

Table 1.14: ANOVA Table for Split Plot Design.

The test for the  $U$  factors is  $\frac{MSE_U}{MSE_{R \times U}}$  against  $F_{1-\alpha, m-1, (m-1)(l-1)}$ . The test for the  $C$  factors is  $\frac{MSE_C}{MSE_{\text{Error 2}}}$  against  $F_{1-\alpha, n-1, (n-1)m(l-1)}$ . The test for the  $U \times C$  interactions is  $\frac{MSE_{U \times C}}{MSE_{\text{Error 2}}}$  against  $F_{1-\alpha, (n-1)(m-1), (n-1)m(l-1)}$ .

When there is no replication  $l-1=0$  so there is no degrees of freedom to estimate  $\sigma_s^2 + n\sigma_w^2$ . By the same reasoning  $\sigma_s^2$  is unable to be estimated as well. In this case two QQPlots can be drawn. One for the whole plot effects and one for the split plot effects. From these the significant effects can be found. If the design for the whole plot or split plot are factorial designs then the higher order interactions can be deemed insignificant and their sums of squares pooled to give an estimate of the errors in the plot or subplot. However this can lead to biased results.

If a  $U \times C$  interaction is significant it means that the control factor is a dispersion effect. Any control factors that have significant main or interaction effects with other control factors are mean control factors.

### Example

In this contrived example the amount of wear on anodes is to be analysed. There are three controllable factors

1. shape of the anode; regular and rounded,

2. tar content; usual and extra, and
3. coke content; usual and extra.

There are also three uncontrollable factors which are

1. voltage; usual and extra,
2. computer controller; new or old and
3. temperature; high and low.

The data was collected by the way of a split plot design with a pot being a wholeplot and an anode being a subplot. The uncontrollable factors were arranged in a  $F2^3$  design with 8 pots being necessary. The controllable factors were also arranged in a  $F2^3$  design and 8 anodes in a pot were needed. The data and design matrix is in appendix H. This was used to get the effects by least squares.

The whole plot had 3 factors and  $2^3 = 8$  treatments so three main effects, three two-way interactions and a three-way interaction are able to be found. This adds up to seven effects. Looking at the ANOVA table there are only  $m - 1 = 8 - 1 = 7$  degrees of freedom associated with the whole plot therefore no degrees of freedom are available to estimate the error for the whole plot. The whole plot effects were plotted on a QQPlot and the significant effects were the main effect of *temp* and the *volt*  $\times$  *temp* interaction. See Figure 1.18a).

In the split plot there were three main effects and three two-way interactions associated with the controllable factors and nine two way interactions between the controllable and uncontrollable factors. Further higher order interactions were deemed insignificant and were pooled to give an estimate of the error variance. As pooling can give biased estimates a QQPlot was made of the split plot effects. The significant effects were associated with *coke*, *tar*  $\times$  *temp* and *shape*. See Figure 1.18b).

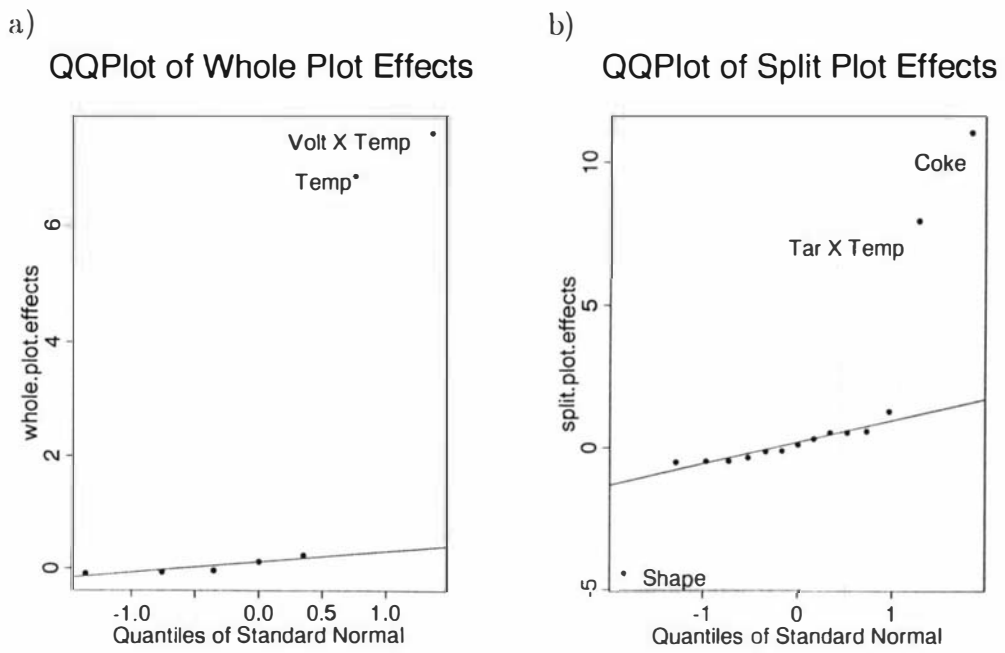


Figure 1.18: QQPlot of a) Whole Plot Effects and b) Split Plot Effects

The effects used for the QQPlot are in Table 1.15

<i>(Intercept)</i>	-0.024435292
Whole Plot Effects	
<i>volt</i>	-0.056244830
<i>cont</i>	-0.028085566
<i>temp</i>	6.887370120
<i>volt : cont</i>	0.235066655
<i>volt : temp</i>	7.638473966
<i>cont : temp</i>	0.122329345
<i>volt : cont : temp</i>	-0.078367673
Split Plot Effects	
<i>shape</i>	-5.170959307
<i>tar</i>	0.120246167
<i>coke</i>	10.035556172
<i>shape : tar</i>	-0.005011248
<i>shape : coke</i>	0.159066978
<i>shape : volt</i>	0.066287725
<i>shape : cont</i>	0.004941601
<i>shape : temp</i>	-0.057336007
<i>tar : coke</i>	-0.099856759
<i>tar : volt</i>	0.017619564
<i>tar : cont</i>	0.226506112
<i>tar : temp</i>	4.908526862
<i>coke : volt</i>	0.178741544
<i>coke : cont</i>	-0.034751595
<i>coke : temp</i>	-0.248175007

Table 1.15: The Effects for the Split Plot Example

The ANOVA table is listed in Table 1.16 and this confirms the significant effects found by the QQPlot at the 5% level.



Table 1.16: Analysis of Variance Table

	<i>Df</i>	<i>Sum of Sq</i>	<i>Mean Sq</i>	<i>F Value</i>	<i>Pr(F)</i>
WholePlot					
U					
<i>temp</i>	1	3035.90			
<i>volt : temp</i>	1	3734.16			
<i>volt</i>	1	0.20			
<i>cont</i>	1	0.05			
<i>volt : cont</i>	1	3.54			
<i>cont : temp</i>	1	0.96			
<i>volt : cont : temp</i>	1	0.39			
<i>whole plot error</i>	0				
SplitPlot					
C					
<i>shape</i>	1	1711.28	1711.28	1243.36	0.000000
<i>coke</i>	1	6445.59	6445.59	4683.16	0.000000
<i>tar</i>	1	0.93	0.93	0.67	0.416973
<i>shape : tar</i>	1	0.00	0.00	0.00	0.972906
<i>shape : coke</i>	1	1.62	1.62	1.18	0.284393
<i>tar : coke</i>	1	0.64	0.64	0.46	0.499741
C × U					
<i>tar : temp</i>	1	1541.99	1541.99	1120.36	0.000000
<i>shape : volt</i>	1	0.28	0.28	0.20	0.653634
<i>shape : cont</i>	1	0.00	0.00	0.00	0.973282
<i>shape : temp</i>	1	0.21	0.21	0.15	0.697836
<i>tar : volt</i>	1	0.02	0.02	0.01	0.904951
<i>tar : cont</i>	1	3.28	3.28	2.39	0.130134
<i>coke : volt</i>	1	2.04	2.04	1.49	0.229864
<i>coke : cont</i>	1	0.08	0.08	0.06	0.813856
<i>coke : temp</i>	1	3.94	3.94	2.86	0.09817
<i>split plot error</i>	41	56.43	1.38		

(based on higher  
order interactions)

The model found is

$$\begin{aligned} \text{wear}|U, C = & 10.04 \times \text{coke} - 5.17 \times \text{shape} + 6.88 \times \text{temp} \\ & + 7.64 \times \text{volt} \times \text{temp} + 4.91 \times \text{tar} \times \text{temp} \end{aligned}$$

The controllable factors *shape* and *coke* enter into the equation only in a linear manner so they are mean control factors. The controllable factor *tar* enters into the equation only as an interaction with the uncontrollable factor *temp* and so *tar* is a dispersion control factor. To make wear robust to the uncontrollable factors then the variance of the mean response of *tar* at one level over both the levels of *temp* should be compared with the other level of *tar*. The interaction graph for these two factors is in Figure 1.19. From the interaction graph it can be seen the required behaviour occurs when when  $\text{tar} = -1$ . However it may also be worthwhile doing further experimentation with lower values of *tar*.

Interaction Plot of Tar and Temp

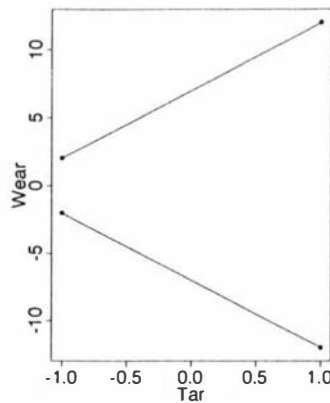


Figure 1.19: Interaction Plot of Tar and Temp

Suppose that the factors were quantitative and that controllable factors are required to be set to give a target value of  $\text{wear} = 5$ . In order to do this some distribution assumptions need to be made about the uncontrollable factors. If we assume that the uncontrollable factors are independent and have expectation zero then

$$E(\text{wear}|UC) = 10.04 \times \text{coke} - 5.17 \times \text{shape} \quad (1.14)$$

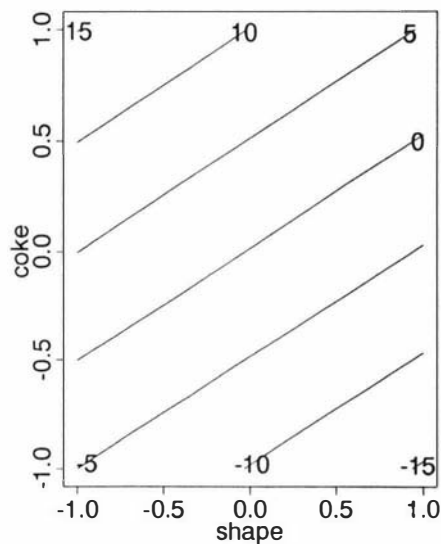
so

$$5 = 10.04 \times \text{coke} - 5.17 \times \text{shape}$$

A contour plot of equation 1.14 was drawn and is in Figure 1.20. Hunting out the  $wear = 5$  contour gives some ideas about the levels of coke and shape that will give the target value. Solving for coke gives

$$coke = .50 + .51 \times shape$$

a) Contour Plot of Coke and Shape



b)

3-D Plot of Coke and Shape versus Wear

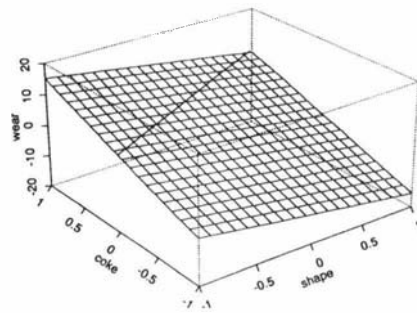


Figure 1.20: a) Contour Plot and b) Perspective Plot.

In the perspective plot the line  $coke = 0.51 \times shape + 0.50$  has been added.

To set the process on target of five values for  $coke$  and  $shape$  conforming to  $coke = .51 \times shape + 0.50$  are chosen for  $shape \in [-1, 1]$  and  $coke \in [0, 1]$ .

Amongst this subset some value of  $shape$  and  $coke$  may be more economical and the process would be run at these values.

In fact  $shape$  is qualitative and not a quantitative factor and can only be set at the values  $\pm 1$ . Setting  $shape = -1$  requires  $coke = -0.01$ . Setting  $shape = 1$  requires that  $coke = 1.01$  which is just outside the experimental region.

### Comments

This method requires that the uncontrollable factors are explicitly set in the experimental design, that there is normality of errors or that a transformation of the response will give normality of errors.

Box and Jones[8] showed that the precision of the estimates derived from the whole plots have the lower precision than those derived from split plots. The precision of the estimates for a completely randomised design and a randomised block design fall between the two. The precision in the estimates for the uncontrollable factors i.e. the noise factors, is not critical as their parameter values are of no consequence, it is their interaction with the split plot factors i.e. control factors, that is of greater importance. The latter have the greater precision as they are estimated in the split plot. However due to physical constraints it may only be possible to have noise factors appearing in the split plot. The disadvantage of this is that the control factors that are now estimated with the least precision and this may reduce the accuracy of the model used for bringing the process to the target value. The important  $U \times C$  interactions are still estimated within the split plot. The ANOVA table for this design is similar to Table 1.14 but with the  $U$ 's and  $C$ 's and their degrees of freedom interchanged.

### 1.9.5 Generalised Linear Models

The aim in parameter design is to minimise the variance of a process while adjusting the mean of the process to some preferred level. However the mean and variance may be related in some way depending on the nature of the data.

Generalised Linear Models gives a way of modelling this relationship between the variance and the mean. It states that  $var(y_i) = \phi_i V(\mu_i)$  when  $y_i$  takes a distribution in the Exponential Family.  $\phi$  is called the dispersion and is that part of the variance that does not depend on the mean of  $y_i$ .

McCullagh and Nelder [21] [25] proposed setting up two GLM models with the exploratory variables being the experimental factors. There is one model for the mean of the process and one for the dispersion with the variance relationship given above connecting the two models.

The definitions for this joint model is

- a set of response variables,  $y_1, \dots, y_n$  which belong to a subfamily of the exponential family,

- a set of parameters  $\beta$  and explanatory variables  ${}_n\mathbf{X}_p = \begin{pmatrix} \mathbf{x}'_1 \\ \vdots \\ \mathbf{x}'_n \end{pmatrix}$ ,

- a differentiable monotonic link function  $g$  such that  $g(\mu_i) = \eta_i = \mathbf{x}'_i\boldsymbol{\beta}$  where  $\mu_i = E(y_i)$ ,
- a variance function  $\text{var}(y_i) = \phi_i V(\mu_i)$ ,
- a set of generalised residuals,  $d_1, \dots, d_n$ ,
- a set of parameters  $\boldsymbol{\gamma}$  and explanatory variables  ${}_n\mathbf{U}_q = \begin{pmatrix} \mathbf{u}'_1 \\ \vdots \\ \mathbf{u}'_n \end{pmatrix}$ ,
- a differentiable monotonic link function  $h$  such that  $h(\phi_i) = \zeta_i = \mathbf{u}'_i\boldsymbol{\gamma}$  where  $\text{var}(d_i) = \tau V_D(\phi)$ .

${}_n\mathbf{U}_q$  and  ${}_n\mathbf{X}_p$  are a subset of the matrix of independent variables.

The extended quasi-likelihood,  $Q^+$  gives a measure of how well this joint model is performing. It is an alternative to the likelihood function when the underlying probability function is not known and relies only on the first and second moments. The extended quasi-deviance, EQD, is defined as

$$-2Q^+ = \sum_{i=1}^n \frac{d_i}{\phi_i} + \sum_{i=1}^n \log [2\pi\phi_i V(y_i)].$$

The  $V(\mu)$  that describes the data best amongst a set of possible  $V(\mu)$  is the one that gives a minimum value for the EQD or a maximum for  $Q^+$ .

The deviance residuals,  $r_D$ , are defined as

$$r_D = \text{sign}(y_i - \mu_i) \sqrt{d_i}$$

$$\text{where } d_i = \int_{\mu_i}^{y_i} \frac{y_i - \mu_i}{V(t)} dt$$

The deviance residuals are distributed approximately normally so the  $d_i$  are distributed approximately  $\phi^2 \chi_1^2$  and a Gamma distribution is used to model them.

## The Method

To fit the model described above two GLM models are fitted simultaneously with one model depending on the other. The first model, the mean model, uses the experimental data to model the means and uses the  $\frac{1}{\phi_i}$ 's as weights. The squared deviance residuals derived from this model are used as the response variable in the second, dispersion model. The fitted values from the latter model form the

$\hat{\phi}_i$ 's. This interlinking of the models, i.e. the weights of the first model are found from the second model and the response variables for the second are found from the first, is resolved by setting  $\frac{1}{\hat{\phi}_i} = 1$  then iterating both models alternately until the weights, and hence the EQD, converges.

For a saturated mean model the process of iteration is unnecessary as a change in weightings does not change the values of any of the parameters in the EQD function or any of the parameter estimates in the two models.

As in the graphical method the idea is to force the data to have a model that is parsimonious i.e. as simple as possible, and separable where the mean and dispersion are not related to each other. Here instead of using one transformation to achieve both criteria, two manipulations are used. To achieve separation an appropriate variance function is chosen which is labelled  $V(\mu)$ . This function ensures that the  $\phi_i$ 's are independent of the mean. The mean-variance plot described in Section 1.9.2 is used to find how the mean and variance are related and hence choose a distribution having the appropriate mean-variance relationship. Parsimony is achieved by choosing the right link function. The Lambda plot in Section 1.9.2 give a useful exploratory tool for working out the types of transformations that the link functions could take.

The four steps in the method are

- Step 1. At the first step  $V(\hat{\mu})$  is found. GLM models for the mean and dispersion model are fitted. For the dispersion model the error distribution is taken to be Gamma with a log link function. At this stage the exact specification of link functions is unimportant although it should conform to the nature of the data i.e. a log link function is inappropriate if the data are not strictly positive. A saturated model for the mean is fitted. For a saturated mean model, the fitted values,  $\hat{\mu}_i$ , remain unchanged for different  $V(\hat{\mu})$  whereas the  $d_i$  do not. The EQD is a function of  $d_i$ 's and  $\hat{\phi}_i$ 's and hence a function of  $V(\hat{\mu})$ . This means it gives a measure of the performance of the  $V(\hat{\mu})$ . The  $V(\hat{\mu})$  that gives the smallest EQD value is taken to be an appropriate  $V(\hat{\mu})$  for the mean model.
- Step 2. In this step a parsimonious model for the dispersions is to be found. The saturated mean model is fitted once again with the  $V(\hat{\mu})$  function found in the last step. The squared deviance residuals are used to model the dispersion. Different link functions for the dispersion model are tried to

see which gives the most parsimonious model. Step-wise analysis of deviance fitting can be used, or as likelihood functions give estimates that are asymptotically Normally distributed, a QQPlot can be used to see with which link function the effects are behaving Normally. However the latter method relies on there being few significant effects.

Another option, if the appropriate link function is thought to be a power function, is to use a lambda plot as described in Section 1.9.2. However instead of using a least squares fit and the t-values, a GLM model is used and t-values of the effects are plotted against  $\lambda$  where the link function is  $\text{Power}(\lambda) = y^\lambda$ .

- Step 3. At this step a parsimonious model for the mean is to be found. The mean model is fitted using several link functions and similarly to Step 2, step-wise analysis of deviance fitting or QQPlotting can be used to find the parsimonious model. A lambda plot can also be used here.
- Step 4. At this stage the parameters of the model are to be estimated. At the last two stages the effects that are significant have been found. We reduce both models from being saturated to only including the parsimonious effects. Now that the mean model is not saturated the weights that are given to it are going to have an effect. The mean model is fitted with weights of one, then the dispersion model is fitted using the resulting  $d_i$ 's. The fitted values give  $\hat{\phi}_i$ . The EQD is calculated. The mean model is fitted again with weights  $\frac{1}{\hat{\phi}_i}$  and so on till the EQD converges. The parameters come from the resulting output. The EQD's can be used to test how well a reduced model fits the data.
- Step 5. The model is then checked for consistency. See [21] chapter 11.

### An Example

Consider again the data in the example in Section 1.9.2 which is to be used as an example for this method. The method is set out here using Splus but packages such as Genstat and GLIM can also be used to do this method. The data is y.ex, the data frame (experimental design) is ex.df.

- Step 1. The mean model is specified as a quasi distribution with an identity

link. The dispersion model is specified as a Gamma distribution with a log link.

From Section 1.9.2 the mean-variance relationship for this data was  $\hat{\sigma}^2 \propto \hat{\mu}^{1.6}$  and the square-root transformation brought parsimony. The Splus code below uses the former information and fits a saturated model for the mean and dispersion model, with the former having an identity link and  $V(\mu) = \mu^2$  and the latter having a log link and Gamma errors. The last line gives the EQD for  $V(\mu) = \mu^2$ . To find the EQD for a different  $V(\mu)$  the variance must be changed in the mean model and  $V(y_i)$  in the EQD equation.

```
mean.model <- glm(y.ex ~ ., data = ex.df, family =
                 quasi(link = identity, variance = "mu^2"),
                 start = rep(100,96), maxit = 300)
deviance.resid <- resid(mean.model)^2
dispersion.model <- glm(deviance.resid ~ ., family =
                       Gamma(link = log), data = ex.df, maxit = 300)
phi <- fitted(dispersion.model)
EQD <- sum(deviance.resid/phi) +
      sum(log(2 * pi * phi * y.ex^2))
```

Several different  $V(\mu)$  were tried. In the Table 1.17 the relevant EQD's are listed.

$V(\mu)$	EQD
1	673.40
$\mu$	673.30
$\mu^2$	673.31
$\mu^3$	673.41

Table 1.17: Extended quasi deviances for different  $V(\mu)$ .

The minimum EQD occurs when  $V(\mu) = \mu$  and this is what will be used in the following steps. Splus is restricted in this case as it only allows  $V(\mu) = \mu^\alpha$  for  $\alpha = 0, 1, 2$  and  $3$  in the GLM function. It is possible to work



out where the minimum in the EQD occurs for values of  $\alpha$  other than these but other than for interest this knowledge is not much use as it can't be used in the modelling process.

The graph in Figure 1.21 is a plot of EQD against  $\alpha$  and shows that for  $\alpha = 1.45$  the EQD is a minimum. This compares to the value of  $\alpha = 1.64$  found by the mean-variance plot i.e.  $\hat{\sigma}^2 \propto \hat{\mu}^{1.64}$ .

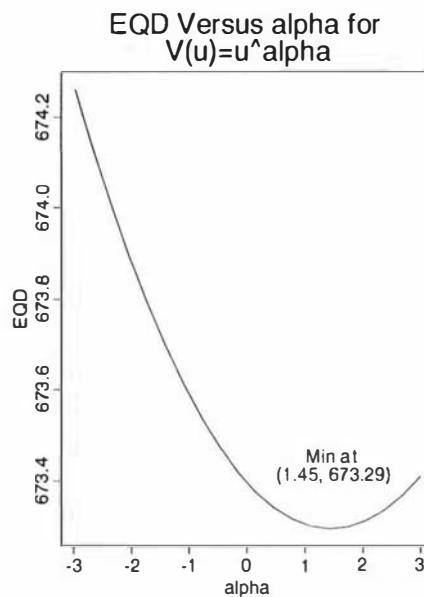


Figure 1.21: A Graph of E.Q.D. for different values of  $\alpha$  for the variance function  $V(\mu) = \mu^\alpha$ .

- Step 2. Now a link function for the dispersion model is to be found that will give parsimony. The Splus code for a identity link for the dispersion model is

```
mean.model <- glm(y.ex ~ ., data = ex.df, family =
                 quasi(link = identity, variance = "mu"),
                 start = rep(100, 96), maxit = 300)
deviance.resid <- resid(mean.model)^2
dispersion.model <- glm(deviance.resid ~ ., family =
                       Gamma(link = identity), data = ex.df,
                       maxit = 300)
```

QQplots for the inverse, identity, square root and log link were QQplotted in Figure 1.22 a), b) c) and d) respectively.

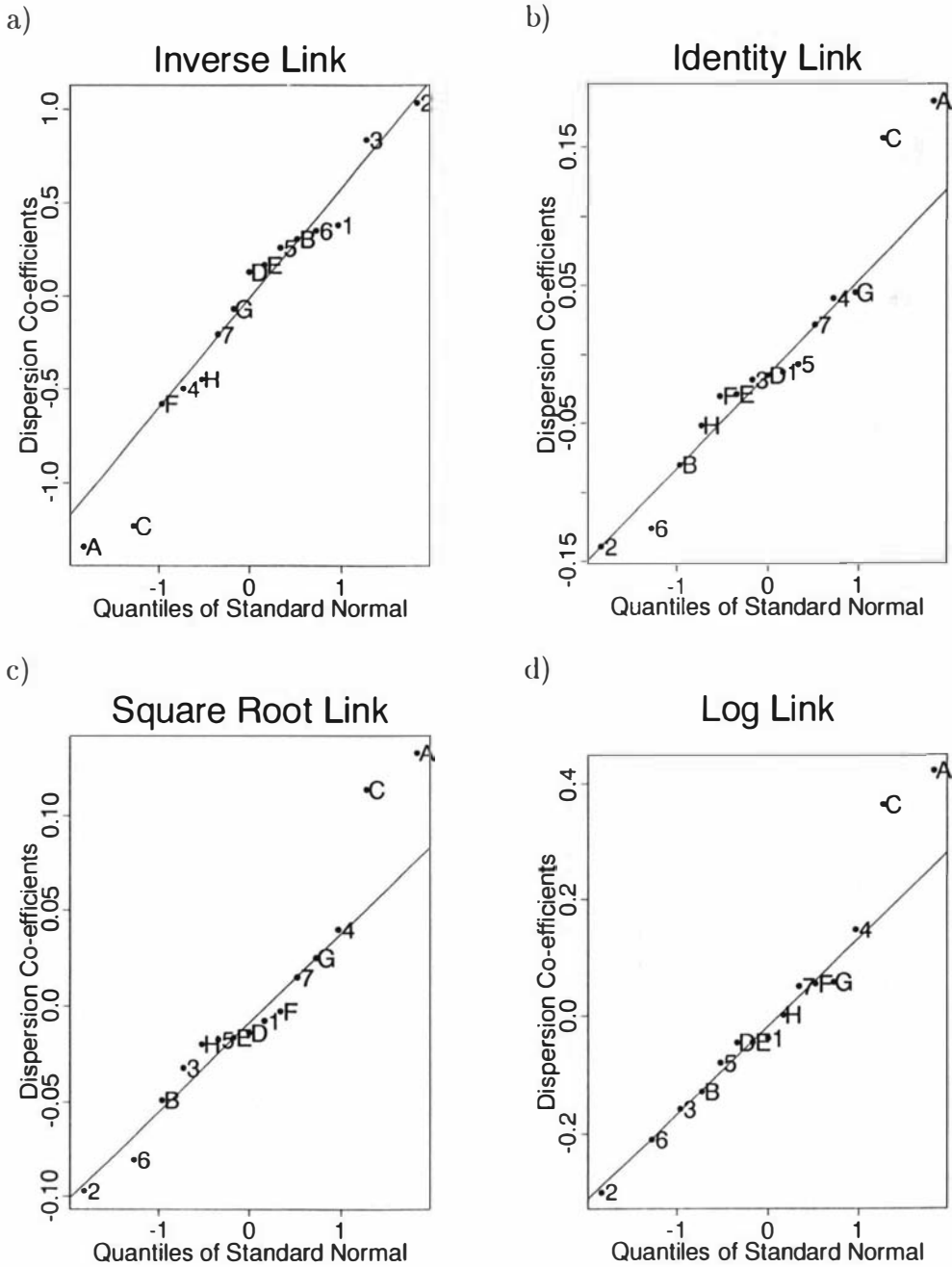


Figure 1.22: QQplots of coefficients for a) Inverse link and b) Identity link c) Square root link and d) Log link.

All of the plots in Figure 1.22 show that A and C are significant factors.

Figures 1.22 a) b) and c) have the insignificant coefficients forming a bowed line instead of a straight line therefore the straight line of 1.22 d), corresponding to a log link function, is preferred.

The analysis of extended quasi deviances for the comparisons between a full model, a model with factors A and C, a model with one each of factors A and C and the null model appear in Table 1.18. There are 96 observations and fitting a saturated mean model reduces the degrees of freedom available for the dispersion model to  $96-15-1=80$ . The degrees of freedom for the null model is 79 and for the saturated model, 64.

Table 1.18: *EQD's* for the Dispersion Model.

Model	<i>EQD</i>	DOF
NULL	695.36	79
A	689.51	78
C	691.164	78
A + C	683.44	77
Full	673.30	64

The difference in *EQD's* between the full model and the A+C model is  $683.44-673.30=10.14$  on  $77-64=13$  degrees of freedom. The reduced model therefore fits the data almost as well as the full model. The difference between *EQD's* for the NULL and A+C model is 11.92 on two degrees of freedom showing that the A+C model is a significant improvement on the null model. Similarly the A+C model is better than models with either factor by itself.

- Step 3. At this step a link function for the mean model is chosen to achieve parsimony. The Splus code for a square root (power(0.5)<sup>2</sup>) link for the mean model is

```
mean.model <- glm(y.ex ~ ., data = ex.df, family =
  quasi(link = power(0.5), variance = "constant"),
  start = rep(100, 96), maxit = 300)
```

<sup>2</sup>Due to a bug in Splus, the start parameter is necessary when using the power function with the variance parameter in the quasi function. The start values should be chosen carefully as values far away from the optimum may lead the GLM function to find a local optimum.

QQplots for the inverse, identity and log link and square root are QQplotted in Figure 1.23 a), b), c) and d) respectively.

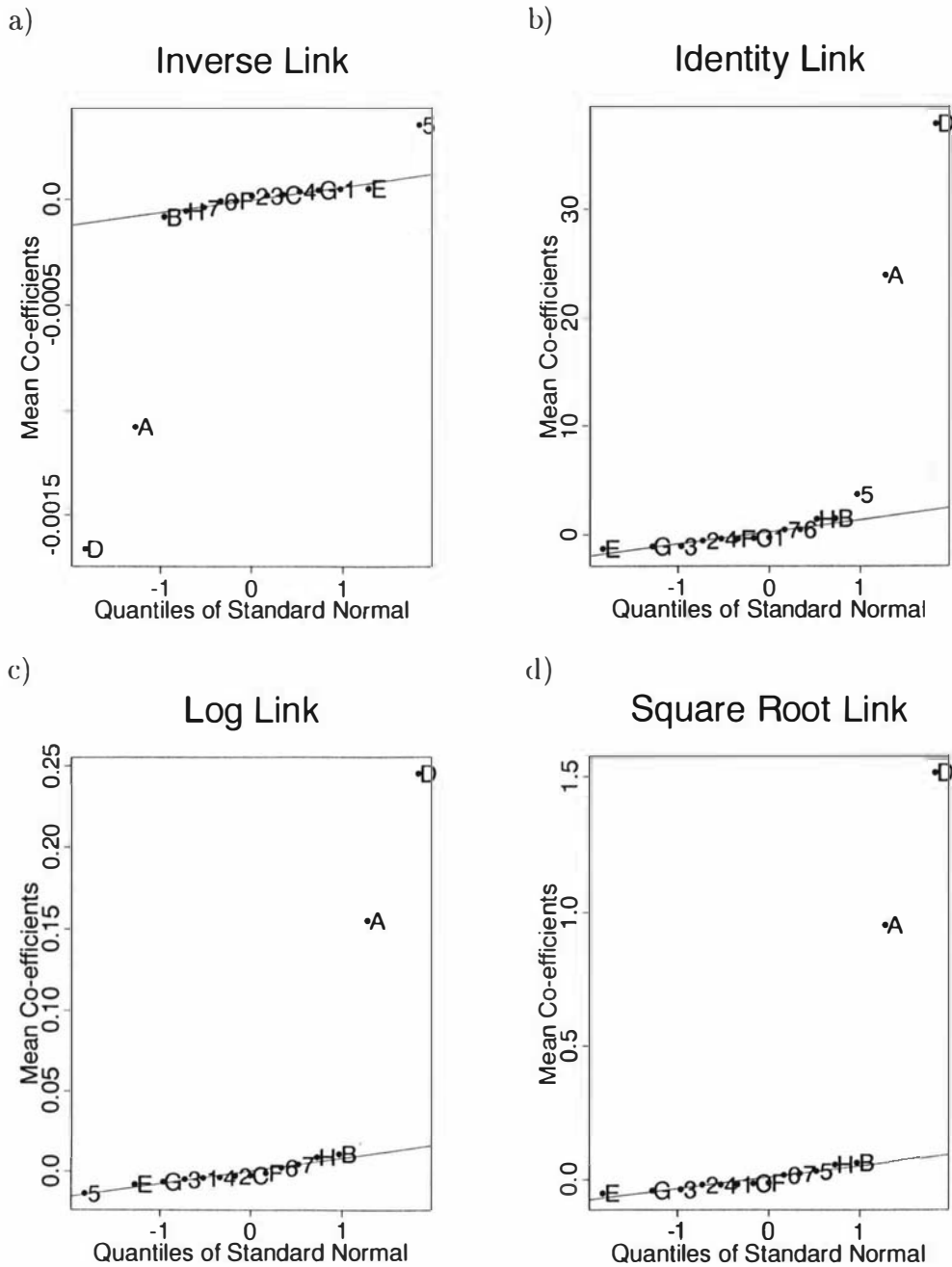


Figure 1.23: QQplots of coefficients for a) Inverse link and b) Identity link c) Log link and d) Square Root link.

All of the plots in Figure 1.23 show that A and D are significant factors. In

Figure 1.23 a) and b) interaction 5, the AD=EH=CG=BF interaction also appears to be significant. Since an interaction model is less parsimonious than a non-interaction model these two links will be discarded. There is little to choose between the log link and square root link of Figures 1.23 c) and d). The lambda plot for mean effects (Figure 1.24 indicates that at  $\lambda \approx 0.5$  the AD interaction passes through zero and the gradients of the A and D effects are approximately zero therefore Power(0.5) is the link function that will be used. The t-values are for the coefficients of the mean model with variance function  $V(\mu) = \mu$  found using the summary function for a GLM object in Splus. Lambda is a value for the power link function i.e.  $\text{Power}(\lambda) = \text{data}^\lambda$ .

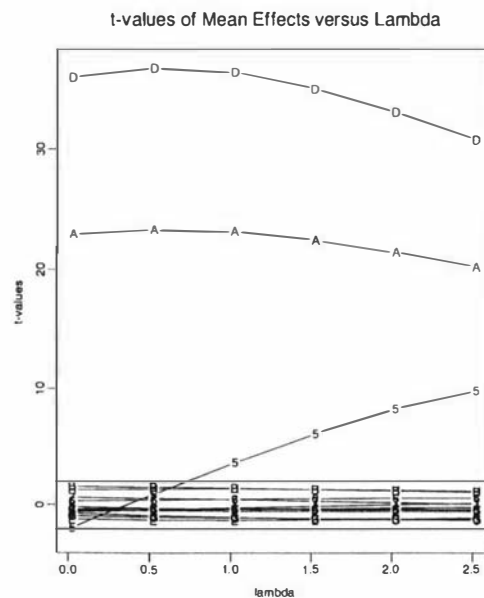


Figure 1.24: Lambda plot of the mean effects using a GLM model and variance function  $V(\mu) = \mu$ .

- Step 4.

At this stage the coefficients of the parameters are wanted. The parsimonious models are known i.e. factors A and D for the mean model and factors A and C for the dispersion model. Due to the non-orthogonality of the effects in GLM models, these factors have to be fitted jointly, without the insignificant effects, to get their parameter values. On reducing the

mean model, the weightings,  $\frac{1}{\phi_i}$ , will change the values of the parameters and the EQD. However, on iteration, the EQD and parameter values quickly converge. In the previous steps this iteration was omitted since in a saturated model for the mean the parameters remain unchanged with changing weights.

The code to do this is listed below. It consists of commands (part 1) that call a function (part 2). Although this may not be good programming style Splus is very inefficient at looping so it is a lot quicker to run part 1 (once with the weighter command and then without) and part 2 ten times than to have a loop iterating for fifty.

- Part 1. This command is entered from the main line

```
weighter<-rep(1,96)
for (i in 1:5) {print(Multi.model.fitter())}
```

- Part 2.

```
Multi.model.fitter<-
function()
{
mean.model      <<- glm(y.ex ~ ., data = ex.df[c(1, 4)],
                      family = quasi(link = power(0.5),
                      variance = "mu"), start =
                      rep(100, 96), maxit = 300,
                      weights = weighter)
deviance.residuals<<- resid(mean.model)^2/weighter
dispersion.model <<- glm(deviance.residuals ~ .,
                      family = Gamma(link = log), data =
                      ex.df[c(1, 3)], maxit = 300)
weighter        <<- fitted(dispersion.model)^-1

EQD <<- sum(deviance.residuals/fitted(dispersion.model)) +
sum(log(2 * pi * fitted(dispersion.model) * y.ex))
}
```

All the variables are global since a GLM function which is itself in a

function looks not at the function it is in but at the global frame to find out the value of weights. Having global variables also means that the outputs from the GLM function are accessible from the global level.

Now that the apparatus is set up to do the iteration, the EQD for the reduced mean model can be found, which takes into account the dispersion model with only factors A and C included. The table listing these is Table 1.19

Model	<i>EQD</i>	DOF
NULL	1003.20	92
A	972.70	91
D	866.43	91
A + D	698.18	90
Full	683.44	77

Table 1.19: *EQD*'s for the Mean Model taking into account the Dispersion Model.

From the EQD table the difference between the FULL model and the A+D model is 14.74 on 13 degrees of freedom which means the A+D model is doing a good job in modeling the data. The difference in EQD's between the A+D model and the NULL model is 305 on 2 degrees of freedom showing that the former model is a vast improvement over the null model.

The sequence of code above was used to find the co-efficients for the models with A and C as mean factors and A and D as dispersion factors. The models are

$$\begin{aligned}\sqrt{\hat{\mu}} &= 12.46 + 0.97 \times A + 1.49 \times D \\ \log \hat{\phi} &= -0.59 + 0.33 \times A + 0.34 \times C \\ \text{and } \text{Var}(y_i) &= \phi_i V(\mu_i) \\ &= \exp[-0.59 + 0.33A + 0.34C] \times (12.46 + 0.97A + 1.49D)^2\end{aligned}$$

### Comments

The uncontrollable factors need not be explicitly set in the experiment.

The method described in Section 1.9.2 uses one transformation to achieve parsimony and separation. For some data this may not be achievable in one transformation. In this method separation and parsimony are achieved separately. The former method relies on the transformed data being Normally distributed. This is reasonable when there is a moderate amount of data due to the Central Limit Theorem. In this method Normality of the data is not assumed and so it can be applied to a much wider range of data i.e. counts and proportions. Many of the previous methods relied on the designs being factorial designs with factors at two levels while here factors with more levels and covariate factors can be used.

The method is not well documented and what has been written about the method is at a very high level. Persons unfamiliar with GLM models and appropriate GLM packages for model fitting would find wading through the literature very hard going.

### 1.9.6 Restricted Generalised Linear Models

Engel[13] proposed this method based around the ideas of GLM. The method is more restrictive than the GLM method but has the advantage that all the parameters can be estimated by least squares. Fitting GLM models involves specialist software or specialist knowledge

#### The Model

The combined model for the mean,  $\mu$  and dispersion,  $\phi$  is

$$E(y_i) = \mu_i = \mathbf{x}_i\boldsymbol{\beta}, \quad (1.15)$$

$$\text{var}(y_i) = \phi_i V(\mu_i, \theta), \quad (1.16)$$

$$\log(\phi_i) = \mathbf{z}_i\boldsymbol{\gamma}. \quad (1.17)$$

where  $\mathbf{x}_i$  and  $\mathbf{z}_i$  are subsets of the matrix of independent variables and  $\mu_i$  is the mean at each distinct design point.  $V(\mu_i, \theta)$  is the component of the variance related to the mean and  $\phi_i$  is the component of the variance not related to the mean i.e. the dispersion.

This is similar to the GLM model but in this case the links are fixed. For the mean model (equation 1.15) it is the identity link and for the variance model (equation 1.17) it is the log link. In this model only “power of the mean”



variances are considered i.e.  $V(\mu_i, \theta) = \mu_i^\theta$ .  $\bar{y}_i$  is used to estimate  $\mu$  where  $\bar{y}_i$  is the average value at each distinct design point. Similarly  $s_i^2$  estimates  $var(y_i)$ .

In terms of the attributes of parsimony and separation, separation is achieved by picking the right  $V(\mu_i, \theta)$  which hopefully can be modelled as  $\mu_i^\theta$ . If it is some other variance function then this method may give inaccurate results. Parsimony is achieved by choosing the right link function. In this case they are pre-chosen. If parsimony is not achieved then a transformation of the data can be done and the method of estimation redone. In this case the  $V(\mu_i, \theta)$  function may change.

## The Method

The estimation procedure goes like this

- Step 1. First an estimate of  $\theta$  is found by least squares and with the model.

$$\log s_i^2 = \log \phi + \theta \log \bar{y}_i + error.$$

The aim in this step is to find out what  $\theta$  is for the power of the mean model by using the model specification  $var(y_i) = \phi_i V(\mu_i, \theta)$  where  $V(\mu_i, \theta) = \mu_i^\theta$ .

- Step 2. The significant factors in the matrix of independent variables are found by observing the parameters estimates of  $\gamma$  using least squares and the model

$$\log \left( \frac{s_i^2}{\bar{y}_i^\theta} \right) = \mathbf{X}_i \gamma + error$$

where  $\mathbf{X}$  is the matrix of independent variables. The significant factors are renamed  $\mathbf{z}_i$ .

- Step 3. Using the appropriate significant factors for the dispersion model, which were found in the last step,  $\gamma$  and  $\theta$  are re-estimated via least squares and the model

$$\log(s_i^2) = \mathbf{z}_i \gamma + \theta \log(\bar{y}_i) + error$$

Since a new value of  $\hat{\theta}$  may have been found steps 2 and 3 are iterated until a stable set of the factors are found.

- Step 4. Estimates for  $\beta$  are found using weighted least squares, with the weights given by

$$\hat{w}_i^{-1} = \hat{\phi}_i \bar{y}_i^{\hat{\theta}}$$

where

$$\hat{\phi}_i = \exp[z_i \hat{\gamma}]$$

as specified in the equation (1.17) and estimated in the previous step, and with the model

$$\bar{y}_i = \mathbf{X}_i \boldsymbol{\beta} + \text{error}$$

and the significant factors are renamed  $x_i$ .

### Example

Consider again the data in the example in section 1.9.2 which is to be used as an example for this method. The method is set out here using Splus but many simpler statistical packages with weighted least squares will perform this method for example minitab.

The design matrix is called `ex.mat`. The means are in `ex.mean` and the sample variances are in `ex.s2`.

- STEP 1 Estimate  $\theta$  using the model

$$\log s_i^2 = \log \phi + \theta \log \bar{y}_i + \text{error}.$$

The Splus command and output is

```
ls.print(lsfite(log(ex.mean),c(log(ex.s2))))
```

Residual Standard Error = 0.7533, Multiple R-Square = 0.315

N = 16, F-statistic = 6.4392 on 1 and 14 df, p-value = 0.0237

	coef	std.err	t.stat	p.value
Intercept	-3.8866	3.2549	-1.1941	0.2523
X1	1.6411	0.6467	2.5375	0.0237

This gives  $\hat{\theta}=1.64$  and  $\log \hat{\phi}=-3.89$ . Only the latter is significant at the 5% level.

- STEP 2 The significant factors for the variance model are found using the model

$$\log \left( \frac{s_i^2}{\bar{y}_i^\theta} \right) = \mathbf{X}_i \boldsymbol{\gamma} + \text{error}.$$

All the factors are fitted and then plotted on a QQPlot to find which are significant. The Splus command and output follow and the QQPlot is Figure 1.25.

```
ls.print(lsfite(ex.mat,c(log(ex.s2/ex.mean^1.6411))))
```

```
Residual Standard Error = NA, Multiple R-Square = 1
N = 16, F-statistic = NA on 15 and 0 df, p-value = NA
```

	coef	std.err	t.stat	p.value
Intercept	-3.8866	NA	NA	NA
X1	0.3209	NA	NA	NA
X2	-0.1387	NA	NA	NA
X3	0.3652	NA	NA	NA
X4	-0.2004	NA	NA	NA
X5	-0.0433	NA	NA	NA
X6	0.0569	NA	NA	NA
X7	0.0639	NA	NA	NA
X8	-0.0062	NA	NA	NA
X9	-0.0353	NA	NA	NA
X10	-0.3030	NA	NA	NA
X11	-0.1546	NA	NA	NA
X12	0.1500	NA	NA	NA
X13	-0.0696	NA	NA	NA
X14	-0.2125	NA	NA	NA
X15	0.0452	NA	NA	NA

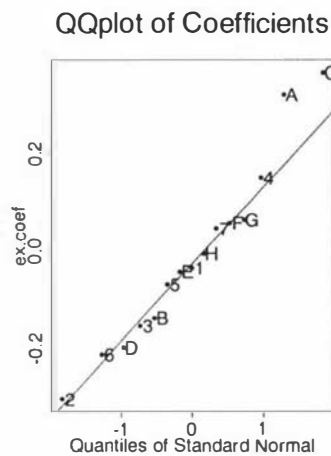


Figure 1.25: QQPlot of Dispersion Effects.

From Figure 1.25 factors  $A$  and  $C$  are the only significant factors.

- STEP 3

Using the smaller set of factors, i.e. factors  $A$  and  $C$ ,  $\gamma$  and  $\theta$  are re-estimated using

$$\log(s_i^2) = z_i \gamma + \theta \log(\bar{y}_i) + \text{error}.$$

```
ls.print(lsfite(cbind(ex.mat[,c(1,3)], log(ex.mean)), c(log(ex.s2))))
```

Residual Standard Error = 0.5425, Multiple R-Square = 0.6955  
 N = 16, F-statistic = 9.1367 on 3 and 12 df, p-value = 0.002

	coef	std.err	t.stat	p.value
Intercept	0.1630	2.7695	0.0589	0.9540
X1	0.4461	0.1603	2.7827	0.0166
X2	0.3632	0.1356	2.6776	0.0201
X3	0.8351	0.5505	1.5169	0.1552

This gives  $\hat{\gamma}_A = 0.4461$ ,  $\hat{\gamma}_C = 0.3632$  and  $\hat{\theta} = 0.8351$ . Since  $\theta$  has changed step 2 is redone to see if the set of significant factors changes. In this case factors  $A$  and  $C$  remain the only significant factors.

- STEP 4.

Using  $\hat{\gamma}_A$ ,  $\hat{\gamma}_C$  and  $\hat{\theta}$  found previously the weights are found i.e.

$$\hat{\phi} = \exp[z\hat{\gamma}] = \exp[\hat{\gamma}_A z_1 + \hat{\gamma}_C z_3]$$

and

$$w^{-1} = \hat{\phi}\bar{y}$$

then the mean model

$$\bar{y}_i = X_i\beta + error$$

is fitted.

```
phi<-exp(0.4461*ex.mat[,1]+0.3632*ex.mat[,3])
```

```
weightinv<-phi*ex.mean^0.8351
```

```
ls.print(lsfit(ex.mat,c(ex.mean),wt=(weightinv^-1)))
```

Residual Standard Error = NA, Multiple R-Square = 1

N = 16, F-statistic = NA on 15 and 0 df, p-value = NA

	coef	std.err	t.stat	p.value
Intercept	158.5325	NA	NA	NA
X1	23.8834	NA	NA	NA
X2	1.4862	NA	NA	NA
X3	-0.3669	NA	NA	NA
X4	37.7849	NA	NA	NA
X5	-1.3259	NA	NA	NA
X6	-0.4160	NA	NA	NA
X7	-1.1242	NA	NA	NA
X8	1.4166	NA	NA	NA
X9	-0.2969	NA	NA	NA
X10	-0.5677	NA	NA	NA
X11	-1.0864	NA	NA	NA
X12	-0.4265	NA	NA	NA
X13	3.6623	NA	NA	NA
X14	0.4870	NA	NA	NA

X15    0.4555    NA    NA    NA

QQPlotting the parameters from the table above in Figure 1.26 shows that factors  $A$  and  $D$  are significant mean effects.

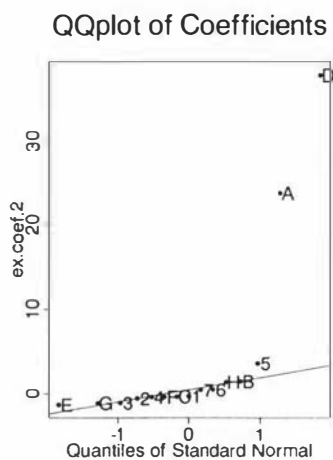


Figure 1.26: QQPlot of Mean Effects.

The parameters for factors  $A$  and  $D$  were re-estimated since with the weightings they lose the property of orthogonality.

```
ls.print(lsfit(ex.mat[,c(1,4)],c(ex.mean),wt=(weightinv^-1)))
```

Residual Standard Error = 0.6933, Multiple R-Square = 0.9876  
N = 16, F-statistic = 517.8323 on 2 and 13 df, p-value = 0

	coef	std.err	t.stat	p.value
Intercept	158.2268	1.4884	106.3043	0
X1	23.4768	1.4662	16.0117	0
X2	35.5061	1.2778	27.7862	0

So  $\beta_A=23.48$  and  $\beta_C=35.51$ .

The conclusion drawn is that  $D$  is a mean effect,  $C$  is a dispersion effect,  $A$  is both and all the other factors are neither. The model for them is

$$\begin{aligned}\hat{\mu}_i &= 158.23 + 23.48A_i + 35.51D_i & (1.18) \\ \log(\hat{\phi}_i) &= 0.45A_i + 0.36C_i \\ \text{var}(y_i) &= \phi_i V(\mu_i, \theta) \\ &= \exp(0.45A_i + 0.36C_i) \times (158.23 + 23.48A_i + 35.51D_i)^{0.84}\end{aligned}$$

To achieve minimum dispersion factors  $A$  and  $C$  would be set to their lowest level and then using equation 1.18 the mean would be brought to the target value through factor  $D$ . If the target is not met then  $A$  will need to be adjusted to meet the target. This will increase the variance.

### 1.9.7 Comments

The uncontrollable factors need not be set explicitly in the experimental design. The only major flaw in this method is that some alternative variance functions are not possible, for example the binomial variance function  $V(\mu_i, \theta) = \mu_i(1 - \mu_i)$  and the negative binomial variance function  $V(\mu_i, \theta) = \mu_i(1 - \theta\mu_i)$ .

It has the advantage over the generalised linear model method of the previous section in that the required steps need a lower level of expertise and computing power to use this method.

■

## Chapter 2

# Optimisation Under Changing Conditions

One of the aims of Parameter Design is to keep the mean response on target while simultaneously minimising the variance. The model for the mean response and variance response is found in terms of the controllable factors. However, when the process is running, some of the uncontrollable factors may be observable. This additional information should enhance our choice of the levels of the controllable factors. What I propose is to split the uncontrollable factors into two sets, observable and unobservable factors. In the first set there may be factors like temperature and humidity while in the second there may be component quality which may be unmeasurable due to expense. The aim will be to find a relationship between the controllable factors and the observable uncontrollable factors while simultaneously minimising the variance of the response and keeping the mean response on target. During the production process whenever a particular value of an observable uncontrollable factors occurs an appropriate value for the controllable factors can be generated, using the relationship discovered, that meets the stated objective.

In practice the type of observable uncontrollable factors that would give the most benefit are the ones that change smoothly over time, like temperature and humidity, or ones that have a slower rate of influence than the production rate. Certainly if an observable factor changes its values in wild swings it would not be sensible making continuous invasive adjustments to the product/process. In such cases it may be advisable to enforce some control over the factor.

---

<sup>1</sup>Selected sections of this chapter were published in a paper that appears in Appendix B



## 2.1 A First Order Model

As an example consider the first order model below which is an extension of the ideas of Myers et al[24]. In this case the model has been generated through Response Surface Designs however other types of experimental designs could lead to a model such as this.

Let  $Y$  represent the quality variable and  $y_i$  a response. Let  $\mathbf{x}_i$  represent the controllable factors,  $\mathbf{t}_i$  the observable uncontrollable factors and  $\mathbf{z}_i$  the unobservable uncontrollable factors.

Let the model for the quality response variable,  $y_i$ , be

$$y_i|\mathbf{x}_i, \mathbf{t}_i, \mathbf{z}_i = \beta_0 + \mathbf{x}_i^t \boldsymbol{\beta}_1 + \mathbf{t}_i^t \boldsymbol{\beta}_2 + \mathbf{z}_i^t \boldsymbol{\beta}_3 + \mathbf{x}_i^t \boldsymbol{\Sigma}_1 \mathbf{t}_i + \mathbf{z}_i^t \boldsymbol{\Sigma}_2 \mathbf{x}_i + \mathbf{z}_i^t \boldsymbol{\Sigma}_3 \mathbf{t}_i + \epsilon_i \quad (2.1)$$

where the response is dependent on  $\mathbf{x}_i$ ,  $\mathbf{t}_i$ , and  $\mathbf{z}_i$ .

The list below defines the sizes of the matrices and vectors.

$$\begin{array}{llll} \beta_0 & 1 \times 1 & & \\ \boldsymbol{\beta}_1 & p \times 1, & \mathbf{x}_i & p \times 1, & \boldsymbol{\Sigma}_1 & q \times p, & \mathbf{t}_i & \sim \text{Dist}(0, \boldsymbol{\Lambda}_1) \\ \boldsymbol{\beta}_2 & q \times 1, & \mathbf{t}_i & q \times 1, & \boldsymbol{\Sigma}_2 & r \times p, & \mathbf{z}_i & \sim \text{Dist}(0, \boldsymbol{\Lambda}_2) \\ \boldsymbol{\beta}_3 & r \times 1, & \mathbf{z}_i & r \times 1, & \boldsymbol{\Sigma}_3 & r \times q, & \epsilon_i & \sim \text{Dist}(0, \sigma_\epsilon^2) \end{array}$$

where  $\text{Dist}(\boldsymbol{\mu}, \boldsymbol{\Lambda})$  represents some joint distribution with  $\boldsymbol{\mu} = \mathbf{0}$  and variance-covariance matrix  $\boldsymbol{\Lambda}$ . The last requirement is that  $\epsilon$ ,  $\mathbf{t}$  and  $\mathbf{z}$  be independent.

### 2.1.1 Model One

The aim here will be to minimise variance of  $Y|\mathbf{x}_i, \mathbf{t}_i$  while keeping the mean of  $Y|\mathbf{x}_i, \mathbf{t}_i$  to target and finding a relationship between the  $\mathbf{x}_i$  and  $\mathbf{t}_i$ . The factors  $\mathbf{z}_i$  are eliminated as the only information they provide are through their means and variances. First the mean of  $Y|\mathbf{x}_i, \mathbf{t}_i$  will be found

**Theorem 2.1**

$$E_Y(Y_i|\mathbf{x}_i, \mathbf{t}_i) = \beta_0 + \mathbf{x}_i^t \boldsymbol{\beta}_1 + \mathbf{t}_i^t \boldsymbol{\beta}_2 + \mathbf{x}_i^t \boldsymbol{\Sigma}_1 \mathbf{t}_i$$

**Proof**

$$\begin{aligned}
 E_Y(Y_i|\mathbf{x}_i, \mathbf{t}_i) &= E_Z(E_Y(Y_i|\mathbf{x}_i, \mathbf{t}_i, \mathbf{z}_i)) \\
 &= E_Z(\beta_0 + \mathbf{x}_i^t \beta_1 + \mathbf{t}_i^t \beta_2 + \mathbf{z}_i^t \beta_3 + \mathbf{x}_i^t \Sigma_1 \mathbf{t}_i + \mathbf{z}_i^t \Sigma_2 \mathbf{x}_i + \mathbf{z}_i^t \Sigma_3 \mathbf{t}_i) \\
 &= \beta_0 + \mathbf{x}_i^t \beta_1 + \mathbf{t}_i^t \beta_2 + \mathbf{x}_i^t \Sigma_1 \mathbf{t}_i
 \end{aligned}$$

■

and then the variance of  $Y|\mathbf{x}_i, \mathbf{t}_i$ .

**Theorem 2.2**

$$Var_Y(Y_i|\mathbf{x}_i, \mathbf{t}_i) = \sigma_\epsilon^2 + (\beta_3 + \Sigma_2 \mathbf{x}_i + \Sigma_3 \mathbf{t}_i)^t \Lambda_2 (\beta_3 + \Sigma_2 \mathbf{x}_i + \Sigma_3 \mathbf{t}_i)$$

**Proof**

$$\begin{aligned}
 Var_Y(Y_i|\mathbf{x}_i, \mathbf{t}_i) &= E_Z(Var_Y(Y_i|\mathbf{x}_i, \mathbf{t}_i, \mathbf{z}_i)) + Var_Z(E_Y(Y_i|\mathbf{x}_i, \mathbf{t}_i, \mathbf{z}_i)) \\
 &= \sigma_\epsilon^2 + Var_Z(\beta_0 + \mathbf{x}_i^t \beta_1 + \mathbf{t}_i^t \beta_2 + \mathbf{x}_i^t \Sigma_1 \mathbf{t}_i + \mathbf{z}_i^t (\beta_3 + \Sigma_2 \mathbf{x}_i + \Sigma_3 \mathbf{t}_i)) \\
 &= \sigma_\epsilon^2 + (\beta_3 + \Sigma_2 \mathbf{x}_i + \Sigma_3 \mathbf{t}_i)^t \Lambda_2 (\beta_3 + \Sigma_2 \mathbf{x}_i + \Sigma_3 \mathbf{t}_i)
 \end{aligned}$$

■

Now we wish to minimise  $Var_Y(Y_i|\mathbf{x}_i, \mathbf{t}_i)$  subject to  $T = E_Y(Y_i|\mathbf{x}_i, \mathbf{t}_i)$  where  $T$  is the target.

**Theorem 2.3** *If  $\Lambda_2$  is positive definite and  $\Sigma_2$  is of rank  $r$  then the value that minimises*

$$Var_Z(E_Y(Y_i|\mathbf{x}_i, \mathbf{t}_i, \mathbf{z}_i)) = \sigma_\epsilon^2 + (\beta_3 + \Sigma_2 \mathbf{x}_i + \Sigma_3 \mathbf{t}_i)^t \Lambda_2 (\beta_3 + \Sigma_2 \mathbf{x}_i + \Sigma_3 \mathbf{t}_i)$$

subject to

$$T = E_Y(Y_i|\mathbf{x}_i, \mathbf{t}_i) = \beta_0 + \mathbf{x}_i^t \beta_1 + \mathbf{t}_i^t \beta_2 + \mathbf{x}_i^t \Sigma_1 \mathbf{t}_i$$

is

$$\mathbf{x}_i^* = (2D^t B)^{-1} (F + 2E^t B) D - E. \quad (2.2)$$

Where

$$\begin{aligned}
 \mathbf{A} &= (\Sigma_2^t \Lambda_2 \Sigma_2)^{-1}, & \text{assuming that } \mathbf{A} \text{ is invertible,} \\
 \mathbf{B} &= \frac{1}{2}(\beta_1 + \Sigma_1 \mathbf{t}_i), & \mathbf{C} = \Sigma_2^t \Lambda_2 (\beta_3 + \Sigma_3 \mathbf{t}_i), \\
 \mathbf{D} &= \mathbf{A}\mathbf{B}, \mathbf{E} = \mathbf{A}\mathbf{C}, \text{ and } & \mathbf{F} = T - \beta_0 - \mathbf{t}_i^t \beta_2.
 \end{aligned}$$

**Proof**

Lagrangian minimisation will be used.

$$L = \sigma_\epsilon^2 + (\beta_3 + \Sigma_2 \mathbf{x}_i + \Sigma_3 \mathbf{t}_i)^t \Lambda_2 (\beta_3 + \Sigma_2 \mathbf{x}_i + \Sigma_3 \mathbf{t}_i) + \lambda (T - \beta_0 - \mathbf{x}_i^t \beta_1 - \mathbf{t}_i^t \beta_2 - \mathbf{x}_i^t \Sigma_1 \mathbf{t}_i)$$

$$\frac{\partial L}{\partial \mathbf{x}_i} = 2 \Sigma_2^t \Lambda_2 \Sigma_2 \mathbf{x}_i + 2 \Sigma_2^t \Lambda_2 \beta_3 + 2 \Sigma_2^t \Lambda_2 \Sigma_3 \mathbf{t}_i - \lambda (\beta_1 + \Sigma_1^t \mathbf{t}_i)$$

$$\text{therefore } \mathbf{x}_i = (\Sigma_2^t \Lambda_2 \Sigma_2)^{-1} \left( \frac{\lambda}{2} (\beta_1 + \Sigma_1^t \mathbf{t}_i) - \Sigma_2^t \Lambda_2 (\beta_3 + \Sigma_3 \mathbf{t}_i) \right)$$

Let  $\mathbf{A} = (\Sigma_2^t \Lambda_2 \Sigma_2)^{-1}$ , assuming that  $\mathbf{A}$  is invertible,  $\mathbf{B} = \frac{1}{2} (\beta_1 + \Sigma_1^t \mathbf{t}_i)$  and  $\mathbf{C} = \Sigma_2^t \Lambda_2 (\beta_3 + \Sigma_3 \mathbf{t}_i)$ , so  $\mathbf{x}_i = \mathbf{A} (\lambda \mathbf{B} - \mathbf{C})$ .

Let  $\mathbf{D} = \mathbf{A} \mathbf{B}$  and  $\mathbf{E} = \mathbf{A} \mathbf{C}$ , therefore  $\mathbf{x}_i = \lambda \mathbf{D} - \mathbf{E}$ .

$$\begin{aligned} \text{Now } \frac{\partial L}{\partial \lambda} &= T - \beta_0 - \mathbf{x}_i^t (\beta_1 + \Sigma_1^t \mathbf{t}_i) - \mathbf{t}_i^t \beta_2 \\ &= T - \beta_0 - 2(\lambda \mathbf{D}^t - \mathbf{E}^t) \mathbf{B} - \mathbf{t}_i^t \beta_2 \end{aligned}$$

$$\text{therefore } \lambda = (2 \mathbf{D}^t \mathbf{B})^{-1} (F + 2 \mathbf{E}^t \mathbf{B})$$

$$\text{where } F = T - \beta_0 - \mathbf{t}_i^t \beta_2$$

$$\text{and } \mathbf{x}_i^* = (2 \mathbf{D}^t \mathbf{B})^{-1} (F + 2 \mathbf{E}^t \mathbf{B}) \mathbf{D} - \mathbf{E}$$

providing  $\mathbf{A}$  is invertible.

$$\text{Also } \frac{\partial^2 L}{\partial \mathbf{x}_i \partial \mathbf{x}_i^t} = 2 \Sigma_2^t \Lambda_2 \Sigma_2$$

As long as  $\Lambda_2$  is positive definite and  $\Sigma_2$  is of rank  $r$  then  $2 \Sigma_2^t \Lambda_2 \Sigma_2$  is positive definite and so  $\mathbf{x}_i^*$  is the minimum critical point. ■

During production the values of the observable uncontrollable factors  $\mathbf{t}_i$  would be used to generate the values of the controllable factors,  $\mathbf{x}_i^*$  using equation (2.2).

### 2.1.2 Model Two

The usual method of parameter design is to treat the  $\mathbf{t}_i$  and  $\mathbf{z}_i$  as one group and to find a value for  $\mathbf{x}_i$  that keeps  $Y$  on target with least variance. The factors  $\mathbf{t}_i$  and  $\mathbf{z}_i$  are eliminated as the only information they now provide is through their means and variances. First the mean of  $Y | \mathbf{x}_i$  will be found

**Theorem 2.4**

$$E_Y(Y_i|\mathbf{x}_i) = \beta_0 + \mathbf{x}_i^t \boldsymbol{\beta}_1$$

**Proof**

$$\begin{aligned} E_Y(Y_i|\mathbf{x}_i) &= E_T(E_Z(E_Y(Y_i|\mathbf{x}_i, \mathbf{t}_i, \mathbf{z}_i))) \\ &= \beta_0 + \mathbf{x}_i^t \boldsymbol{\beta}_1 \end{aligned}$$

■

Finding the variance of  $y|\mathbf{x}_i$  depends on the following result.

**Theorem 2.5**

$$Var(Y) = E_X(E_Z(Var_Y(Y|X, Z))) + E_X(Var_Z(E_Y(Y|X, Z))) + Var_X(E_Y(Y|X))$$

**Proof**

Now it is known that

$$Var(Y) = E_X(Var_Y(Y|X)) + Var_X(E_Y(Y|X))$$

so

$$Var(Y|X) = E_Z(Var_Y(Y|X, Z)) + Var_Z(E_Y(Y|X, Z))$$

therefore

$$Var(Y) = E_X(E_Z(Var_Y(Y|X, Z)) + Var_Z(E_Y(Y|X, Z))) + Var_X(E_Y(Y|X))$$

which gives the result.

■

Now the variance of  $Y|\mathbf{x}_i$  will be found.

**Theorem 2.6**

$$\begin{aligned} Var_Y(Y_i|\mathbf{x}_i) &= \sigma_\epsilon + (\boldsymbol{\beta}_3 + \boldsymbol{\Sigma}_2 \mathbf{x}_i)^t \boldsymbol{\Lambda}_2 (\boldsymbol{\beta}_3 + \boldsymbol{\Sigma}_2 \mathbf{x}_i) + trace(\boldsymbol{\Sigma}_3^t \boldsymbol{\Lambda}_2 \boldsymbol{\Sigma}_3 \boldsymbol{\Lambda}_1) \\ &\quad + (\boldsymbol{\beta}_2 + \boldsymbol{\Sigma}_1^t \mathbf{x}_i)^t \boldsymbol{\Lambda}_1 (\boldsymbol{\beta}_2 + \boldsymbol{\Sigma}_1^t \mathbf{x}_i) \end{aligned} \tag{2.3}$$

**Proof**

$$\begin{aligned} Var_Y(Y_i|\mathbf{x}_i) &= E_T(E_Z(Var_Y(Y_i|\mathbf{x}_i, \mathbf{t}_i, \mathbf{z}_i))) + E_T(Var_Z(E_Y(Y_i|\mathbf{x}_i, \mathbf{t}_i, \mathbf{z}_i))) \\ &\quad + Var_T(E_Y(Y_i|\mathbf{x}_i, \mathbf{t}_i)) \end{aligned}$$

Now

$$E_T(E_Z(\text{Var}_Y(Y_i|\mathbf{x}_i, \mathbf{t}_i, \mathbf{z}_i))) = E_T(E_Z(\sigma_\epsilon^2)) = \sigma_\epsilon^2$$

and

$$\begin{aligned} & E_T(\text{Var}_Z(E_Y(Y_i|\mathbf{x}_i, \mathbf{t}_i, \mathbf{z}_i))) \\ &= E_T(\text{Var}_Z(\beta_0 + \mathbf{x}_i^t \beta_1 + \mathbf{t}_i^t \beta_2 + \mathbf{z}_i^t \beta_3 + \mathbf{x}_i^t \Sigma_1 \mathbf{t}_i + \mathbf{z}_i^t \Sigma_2 \mathbf{x}_i + \mathbf{z}_i^t \Sigma_3 \mathbf{t}_i)) \\ &= E_T((\beta_3 + \Sigma_2 \mathbf{x}_i + \Sigma_3 \mathbf{t}_i)^t \Lambda_2 (\beta_3 + \Sigma_2 \mathbf{x}_i + \Sigma_3 \mathbf{t}_i)) \end{aligned}$$

Now

$$\mathbf{t}_i^t \Sigma_3^t \Lambda_2 \Sigma_3 \mathbf{t}_i = \text{trace}(\Sigma_3^t \Lambda_2 \Sigma_3 \mathbf{t}_i \mathbf{t}_i^t)$$

therefore

$$E_T(\mathbf{t}_i^t \Sigma_3^t \Lambda_2 \Sigma_3 \mathbf{t}_i) = \text{trace}(\Sigma_3^t \Lambda_2 \Sigma_3 \Lambda_1)$$

So

$$\begin{aligned} & E_T((\beta_3 + \Sigma_2 \mathbf{x}_i + \Sigma_3 \mathbf{t}_i)^t \Lambda_2 (\beta_3 + \Sigma_2 \mathbf{x}_i + \Sigma_3 \mathbf{t}_i)) \\ &= (\beta_3 + \Sigma_2 \mathbf{x}_i)^t \Lambda_2 (\beta_3 + \Sigma_2 \mathbf{x}_i) + E_T(\mathbf{t}_i^t \Sigma_3^t \Lambda_2 \Sigma_3 \mathbf{t}_i) + 2(\beta_3 + \Sigma_2 \mathbf{x}_i)^t \Sigma_3 \mathbf{t}_i \\ &= (\beta_3 + \Sigma_2 \mathbf{x}_i)^t \Lambda_2 (\beta_3 + \Sigma_2 \mathbf{x}_i) + E_T(\mathbf{t}_i^t \Sigma_3^t \Lambda_2 \Sigma_3 \mathbf{t}_i) \\ &= (\beta_3 + \Sigma_2 \mathbf{x}_i)^t \Lambda_2 (\beta_3 + \Sigma_2 \mathbf{x}_i) + \text{trace}(\Sigma_3^t \Lambda_2 \Sigma_3 \Lambda_1) \end{aligned}$$

Finally

$$\begin{aligned} \text{Var}_T(E_Y(Y_i|\mathbf{x}_i, \mathbf{t}_i)) &= \text{Var}(\beta_0 + \mathbf{x}_i^t \beta_1 + \mathbf{t}_i^t \beta_2 + \mathbf{x}_i^t \Sigma_1 \mathbf{t}_i) \\ &= \text{Var}(\beta_0 + \mathbf{x}_i^t \beta_1 + \mathbf{t}_i^t (\beta_2 + \Sigma_1^t \mathbf{x}_i)) \\ &= (\beta_2 + \Sigma_1^t \mathbf{x}_i)^t \Lambda_1 (\beta_2 + \Sigma_1^t \mathbf{x}_i) \end{aligned}$$

Hence

$$\begin{aligned} \text{Var}_Y(Y_i|\mathbf{x}_i) &= \sigma_\epsilon + E_T((\beta_3 + \Sigma_2 \mathbf{x}_i + \Sigma_3 \mathbf{t}_i)^t \Lambda_2 (\beta_3 + \Sigma_2 \mathbf{x}_i + \Sigma_3 \mathbf{t}_i)) \\ &\quad + \text{Var}(\beta_0 + \mathbf{x}_i^t \beta_1 + \mathbf{t}_i^t \beta_2 + \mathbf{x}_i^t \Sigma_1 \mathbf{t}_i) \\ &= \sigma_\epsilon + (\beta_3 + \Sigma_2 \mathbf{x}_i)^t \Lambda_2 (\beta_3 + \Sigma_2 \mathbf{x}_i) + \text{trace}(\Sigma_3^t \Lambda_2 \Sigma_3 \Lambda_1) \\ &\quad + (\beta_2 + \Sigma_1^t \mathbf{x}_i)^t \Lambda_1 (\beta_2 + \Sigma_1^t \mathbf{x}_i) \end{aligned}$$

■

Now we wish to minimise  $\text{Var}_Y(Y_i|\mathbf{x}_i)$  subject to  $T = E_Y(Y_i|\mathbf{x}_i)$ .

**Theorem 2.7** *If  $\Lambda_2$  is positive definite and  $\Sigma_2$  is of rank  $r$  or if  $\Lambda_1$  is positive definite and  $\Sigma_1$  is of rank  $q$  then the value that minimises*

$$\begin{aligned} \text{Var}_Y(Y_i|\mathbf{x}_i) &= \sigma_\epsilon + (\boldsymbol{\beta}_3 + \Sigma_2 \mathbf{x}_i)^t \Lambda_2 (\boldsymbol{\beta}_3 + \Sigma_2 \mathbf{x}_i) + \text{trace}(\Sigma_3^t \Lambda_2 \Sigma_3 \Lambda_1) \\ &\quad + (\boldsymbol{\beta}_2 + \Sigma_1^t \mathbf{x}_i)^t \Lambda_1 (\boldsymbol{\beta}_2 + \Sigma_1^t \mathbf{x}_i) \end{aligned}$$

subject to

$$T = E_Y(Y_i|\mathbf{x}_i) = \beta_0 + \mathbf{x}_i^t \boldsymbol{\beta}_1$$

is

$$\mathbf{x}_i^{**} = (\mathbf{K}^t \boldsymbol{\beta}_1)^{-1} (T - \beta_0 + \mathbf{J}^t \boldsymbol{\beta}_1) \mathbf{K} - \mathbf{J} \tag{2.4}$$

Where

$$\begin{aligned} \mathbf{G} &= (\Sigma_2^t \Lambda_2 \Sigma_2 + \Sigma_1 \Lambda_1 \Sigma_1^t)^{-1}, && \text{assuming that G is invertible,} \\ \mathbf{H} &= \Sigma_2^t \boldsymbol{\beta}_3 + \Sigma_1 \boldsymbol{\beta}_2, \quad \mathbf{K} = \mathbf{G} \frac{\boldsymbol{\beta}_1}{2} \quad \text{and} \quad \mathbf{J} = \mathbf{G} \mathbf{H}. \end{aligned}$$

**Proof**

Using Lagrangian minimisation once more

$$L = \mathbf{x}_i^t \Sigma_2^t \Lambda_2 \Sigma_2 \mathbf{x}_i + \mathbf{x}_i^t \Sigma_1 \Lambda_1 \Sigma_1^t \mathbf{x}_i + 2\boldsymbol{\beta}_3^t \Sigma_2 \mathbf{x}_i + 2\boldsymbol{\beta}_2^t \Sigma_1^t \mathbf{x}_i + \lambda(T - \beta_0 - \mathbf{x}_i^t \boldsymbol{\beta}_1)$$

$$\frac{\partial L}{\partial \mathbf{x}_i} = 2\Sigma_2^t \Lambda_2 \Sigma_2 \mathbf{x}_i + 2\Sigma_1 \Lambda_1 \Sigma_1^t \mathbf{x}_i + 2\Sigma_2^t \boldsymbol{\beta}_3 + 2\Sigma_1 \boldsymbol{\beta}_2 - \lambda \boldsymbol{\beta}_1$$

therefore  $\mathbf{x}_i = (\Sigma_2^t \Lambda_2 \Sigma_2 + \Sigma_1 \Lambda_1 \Sigma_1^t)^{-1} (\frac{\lambda}{2} \boldsymbol{\beta}_1 - \Sigma_2^t \boldsymbol{\beta}_3 - \Sigma_1 \boldsymbol{\beta}_2)$

Let  $\mathbf{G} = (\Sigma_2^t \Lambda_2 \Sigma_2 + \Sigma_1 \Lambda_1 \Sigma_1^t)^{-1}$ , providing G is invertible

$$\mathbf{H} = \Sigma_2^t \boldsymbol{\beta}_3 + \Sigma_1 \boldsymbol{\beta}_2, \text{ so } \mathbf{x}_i = \mathbf{G} (\frac{\lambda}{2} \boldsymbol{\beta}_1 - \mathbf{H})$$

Let  $\mathbf{K} = \mathbf{G} \frac{\boldsymbol{\beta}_1}{2}$  and  $\mathbf{J} = \mathbf{G} \mathbf{H}$ , so  $\mathbf{x}_i = \lambda \mathbf{K} - \mathbf{J}$

$$\begin{aligned} \frac{\partial L}{\partial \lambda} &= T - \beta_0 - \mathbf{x}_i^t \boldsymbol{\beta}_1 \\ &= T - \beta_0 - (\lambda \mathbf{K}^t - \mathbf{J}^t) \boldsymbol{\beta}_1 \end{aligned}$$

so  $\lambda = (\mathbf{K}^t \boldsymbol{\beta}_1)^{-1} (T - \beta_0 + \mathbf{J}^t \boldsymbol{\beta}_1)$

and  $\mathbf{x}_i^{**} = (\mathbf{K}^t \boldsymbol{\beta}_1)^{-1} (T - \beta_0 + \mathbf{J}^t \boldsymbol{\beta}_1) \mathbf{K} - \mathbf{J}$

Also

$$\frac{\partial^2 L}{\partial \mathbf{x}_i \partial \mathbf{x}_i^t} = 2\Sigma_2^t \Lambda_2 \Sigma_2 + 2\Sigma_1 \Lambda_1 \Sigma_1^t$$

Since  $\Lambda_2$  and  $\Lambda_1$  are Variance-Covariance matrices they are both positive semi-definite. However, if  $\Lambda_2$  is positive definite and  $\Sigma_2$  is of rank  $r$  then

$$\Sigma_2^t \Lambda_2 \Sigma_2$$

is positive definite and so

$$2\Sigma_2^t \Lambda_2 \Sigma_2 + 2\Sigma_1 \Lambda_1 \Sigma_1^t$$

is positive definite. If, instead,  $\Lambda_1$  is positive definite and  $\Sigma_1$  is of rank  $q$  then

$$\Sigma_1 \Lambda_1 \Sigma_1^t$$

is positive definite and so

$$2\Sigma_2^t \Lambda_2 \Sigma_2 + 2\Sigma_1 \Lambda_1 \Sigma_1^t$$

is positive definite.

Therefore if  $\Lambda_2$  is positive definite and  $\Sigma_2$  is of rank  $r$  or if  $\Lambda_1$  is positive definite and  $\Sigma_1$  is of rank  $q$  then  $\mathbf{x}_i^{**}$  is the minimum critical point. ■

Now for some values of  $\mathbf{t}_i$ ,  $\mathbf{x}_i^{**}$  may predict a value for  $y_i$  that is closer to the target than that predicted for  $\mathbf{x}_i^*$ . To compare the two predictions the statistic  $E((y_i - T)^2)$  will be considered for  $\mathbf{x}_i^*$  and  $\mathbf{x}_i^{**}$ .

**Theorem 2.8** *For model one*

$$E(y_i | \mathbf{x}_i^*, \mathbf{t}_i - T)^2 = \text{Var}(y | \mathbf{x}_i^*, \mathbf{t}_i)$$

**Proof**

Under model one  $T = E(y_i | \mathbf{x}_i^*, \mathbf{t}_i)$  so

$$E(y_i | \mathbf{x}_i^*, \mathbf{t}_i - T)^2 = E(y | \mathbf{x}_i^*, \mathbf{t}_i - E(y | \mathbf{x}_i^*, \mathbf{t}_i))^2 = \text{Var}(y | \mathbf{x}_i^*, \mathbf{t}_i)$$

■

**Theorem 2.9** *For model two*

$$E(y_i | \mathbf{x}_i^{**}, \mathbf{t}_i - T)^2 = \text{Var}(y | \mathbf{x}_i^{**}, \mathbf{t}_i) + (\mathbf{t}_i^t \beta_2 + \mathbf{x}_i^{**t} \Sigma_1 \mathbf{t}_i)^2$$

**Proof**

Under model two  $T = E(y_i | \mathbf{x}_i^{**})$  and

$$E(y_i | \mathbf{x}_i^{**}) = E(y_i | \mathbf{x}_i^{**}, \mathbf{t}_i) - (\mathbf{t}_i^t \beta_2 + \mathbf{x}_i^{**t} \Sigma_1 \mathbf{t}_i)$$

so

$$\begin{aligned}
 E(y_i|\mathbf{x}_i^{**}, \mathbf{t}_i - T)^2 &= E(y_i|\mathbf{x}_i, \mathbf{t}_i - E(y_i|\mathbf{x}_i^{**}))^2 \\
 &= E(y_i|\mathbf{x}_i^{**}, \mathbf{t}_i - E(y_i|\mathbf{x}_i^{**}, \mathbf{t}_i) + \mathbf{t}_i^t \beta_2 + \mathbf{x}_i^{**t} \Sigma_1 \mathbf{t}_i)^2 \\
 &= Var(y_i|\mathbf{x}_i^{**}, \mathbf{t}_i) + (\mathbf{t}_i^t \beta_2 + \mathbf{x}_i^{**t} \Sigma_1 \mathbf{t}_i)^2.
 \end{aligned}$$

■

So far two methods have been specified i.e. using  $\mathbf{x}_i^*$  or  $\mathbf{x}_i^{**}$  exclusively. A third approach is to use either depending on the value of  $\mathbf{t}_i$ . On observing a value of  $\mathbf{t}_i$  calculate  $E(y_i - T)^2$  for  $\mathbf{x}_i^*$  and  $\mathbf{x}_i^{**}$  and choose that corresponding to the least  $E(y_i - T)^2$  value.

In practice the model is specified but the coefficients of the model are unknown and these are found by experimentation. These estimated coefficients are used to give estimates of  $y_i$ ,  $\mathbf{x}_i$ ,  $E(y_i)$  and  $Var(y_i)$  denoted by  $\hat{y}_i$ ,  $\hat{\mathbf{x}}_i$ ,  $\hat{E}(y_i)$  and  $\widehat{Var}(y_i)$  respectively.

## 2.2 Simulation Example

To illustrate these procedures let the underlying model be

$$\begin{aligned}
 y_i|\mathbf{x}_i, \mathbf{t}_i, \mathbf{z}_i &= 100 + 5x_1 + 7x_2 - 4t_1 + 3t_2 - 5z_1 - 6z_2 - 4x_1t_1 - 3x_2t_2 + 6z_1x_1 \\
 &\quad - 7x_2z_2 - 7x_1z_2 + 8t_1z_1 - 8t_2z_2 + \epsilon
 \end{aligned} \tag{2.5}$$

Where  $\epsilon$  is distributed as Normal( $\mu = 0, \sigma^2 = 10$ ) and with a target of 100. The first step in parameter design is to do an experiment with all the factors at fixed levels. This was done with a  $2^{6-1}$  factorial design and the results are in Table 2.1.



Factors						Response
$x_1$	$x_2$	$t_1$	$t_2$	$z_1$	$z_2$	
-1	-1	-1	-1	-1	-1	91.5311
1	-1	-1	-1	-1	1	110.09547
-1	1	-1	-1	-1	1	120.90064
1	1	-1	-1	-1	-1	133.24822
-1	-1	1	-1	-1	1	99.8026
1	-1	1	-1	-1	-1	71.35239
-1	1	1	-1	-1	-1	101.25509
1	1	1	-1	-1	1	78.83396
-1	-1	-1	1	-1	1	106.93987
1	-1	-1	1	-1	-1	134.56421
-1	1	-1	1	-1	-1	138.35204
1	1	-1	1	-1	1	98.02461
-1	-1	1	1	-1	-1	96.73753
1	-1	1	1	-1	1	78.68531
-1	1	1	1	-1	1	92.89983
1	1	1	1	-1	-1	125.50157
-1	-1	-1	-1	1	1	79.88534
1	-1	-1	-1	1	-1	93.51969
-1	1	-1	-1	1	-1	80.66505
1	1	-1	-1	1	1	96.4498
-1	-1	1	-1	1	-1	58.77738
1	-1	1	-1	1	1	96.73602
-1	1	1	-1	1	1	101.59202
1	1	1	-1	1	-1	126.37272
-1	-1	-1	1	1	-1	75.88907
1	-1	-1	1	1	1	87.37854
-1	1	-1	1	1	1	66.28703
1	1	-1	1	1	-1	143.92734
-1	-1	1	1	1	1	93.3468
1	-1	1	1	1	-1	122.44309
-1	1	1	1	1	-1	116.39455
1	1	1	1	1	1	83.3959

Table 2.1: The experimental design and the responses for a  $2^{6-1}$  factorial design in  $x_1$ ,  $x_2$ ,  $t_1$ ,  $t_2$ ,  $z_1$  and  $z_2$ .

The model found was

$$\hat{y} = 100.06 + 4.98x_1 + 6.45x_2 - 3.55t_1 + 3.74t_2 - 4.86z_1 - 6.85z_2 - 3.57x_1t_1 \\ + 6.11x_1z_1 - 6.98x_1z_2 - 2.15x_2t_2 - 7.36x_2z_2 + 8.24t_1z_1 - 8.58t_2z_2$$

The residuals were used to estimate  $\sigma_\epsilon^2$  and  $\hat{\sigma}_\epsilon^2 = 11.56$ .

The two parameter variable  $t_i$  will be taken to have a 2 dimensional multivariate Normal distribution with a mean of 0 and a Variance-Covariance matrix equal to the identity matrix. Similarly for  $z_i$ . The parameter  $\epsilon$  will have a standard Normal distribution. Four simulations are to be done to compare four different methods. 100,000 values will be generated for each of the  $z_i$ ,  $t_i$  and  $\epsilon$  and they will be used for each simulation.

In order to use equation (2.2)  $\Lambda_2$  must be positive definite and  $\hat{\Sigma}_2$  must be of rank  $r$ . Now  $\Lambda_2$  is the identity matrix so it is positive definite and

$$\hat{\Sigma}_2 = \begin{pmatrix} 6.11 & 0 \\ -6.98 & -7.36 \end{pmatrix}$$

is of rank 2 and  $r = 2$ . The conditions are met.

For simulation 1.1 equation (2.2) was used to generate the  $\hat{\mathbf{x}}_i^*$  from the  $\mathbf{t}_i$ . The equations for  $\hat{x}_1$  and  $\hat{x}_2$  in terms of  $\mathbf{t}_i$  were generated by Maple and appear in Table 2.2.

$$\begin{aligned} x1 := & -(-.2499999997 + 1.111111111 t1 + .1746031745 t2 \\ & + .2222222226 t1^2 - .6031746025 t2 t1 - .2729591838 t2^2 \\ & - .06802721120 t2^2 t1 + .1428571429 t2^3) / (.5555555552 \\ & + .2222222224 t1 + .2222222222 t1^2 - .5952380950 t2 \\ & - .3333333334 t2 t1 + .2168367346 t2^2) \end{aligned}$$

$$\begin{aligned} x2 := & (-.1899092969 t2 + 1.253968253 t1 - .3492063483 t1^2 \\ & - .4285714285 t2 t1 - .02125850360 t2^2 - .09070294800 t1^2 t2 \\ & + .1904761904 t2^2 t1 + .2000000000 10^{-9} t2^3 - .1785714284) / ( \\ & .5555555552 + .2222222224 t1 + .2222222222 t1^2 \\ & - .5952380950 t2 - .3333333334 t2 t1 + .2168367346 t2^2) \end{aligned}$$

Table 2.2: The Maple output giving the relationship between the controllable factors,  $\mathbf{x}_i$ , and the observable uncontrollable factors,  $\mathbf{t}_i$ , that minimises the variance and keeps the mean on target.

The  $\hat{\mathbf{x}}_i^*$  and  $\mathbf{t}_i$  were used with the  $\mathbf{z}_i$  and  $\epsilon_i$  to generate the  $\hat{y}_i^*$  using the regression model in equation (2.5). The variance of the resulting  $\hat{y}_i^*$  was  $67.72 \pm 1.10$  and the mean  $100.40 \pm 0.03$ .

The variance of  $\hat{y}_i^*$  can be broken into three components. That due

- 1) to  $\epsilon$ ,
- 2) to the influence of the random variable  $\mathbf{z}$  and its interaction with  $\mathbf{x}$  and  $\mathbf{t}$ , and
- 3) to the use of the estimated coefficients.

The size of the latter can be estimated by generating  $y_i$  with  $\epsilon_i$  and  $\mathbf{z}_i$  equalling 0 and keeping  $\hat{\mathbf{x}}_i^*$  and  $\mathbf{t}_i$  as before. The resulting variance was  $1.274 \pm 0.003$ . This variance would decrease with the increased accuracy of estimating the coefficients.

In order to use equation (2.4)  $\Lambda_2$  must be positive definite and  $\hat{\Sigma}_2$  must be of rank  $r$  or  $\Lambda_1$  must be positive definite and  $\hat{\Sigma}_1$  must be of rank  $q$ . It has been shown previously that  $\Lambda_2$  is positive definite and  $\hat{\Sigma}_2$  is of rank  $r$ .  $\Lambda_1$  is the identity matrix so it is positive definite. Now

$$\hat{\Sigma}_1 = \begin{pmatrix} -3.57 & 0 \\ 0 & -2.15 \end{pmatrix}$$

which is of rank 2 and  $q = 2$ . So both the conditions hold.

For the second simulation, simulation 1.2, equation (2.4) was used to give  $\hat{\mathbf{x}}_i^{**} = (0.0358, -0.0363)$  which gives, using equation (2.3),  $\widehat{Var}(y) = 249.40$ . The underlying model is known in this case so  $Var(y)$  can be calculated and is  $Var(y) = 223.67$ . Using  $\hat{\mathbf{x}}_i^{**}$  and the previous  $t_i$  and  $z_i$ , the  $\hat{y}_i^{**}$  were generated from the regression model in equation (2.5) to give a sampling variance of  $223.74 \pm 1.43$  and a mean of  $99.97 \pm 0.05$ .

By accounting for the specific values of the  $t_i$  when choosing the  $\hat{\mathbf{x}}_i$  an approximately 70% decrease in variance was achieved. Graphing the two sets of data in histograms in Figure 2.1 illustrates the reduction in the variance of the response  $\hat{y}_i$  from simulation 1.2 to simulation 1.1.

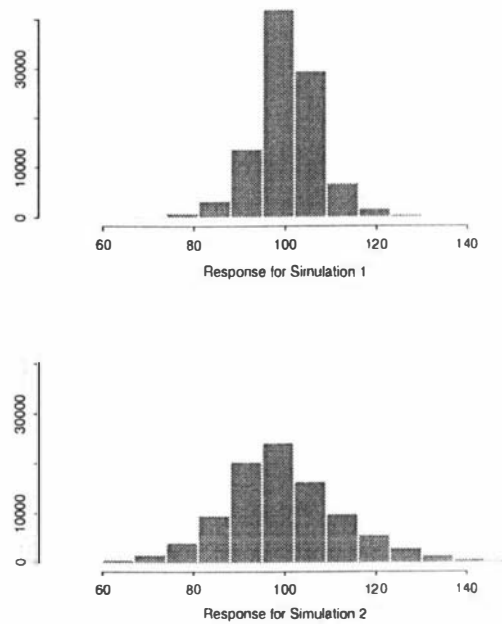


Figure 2.1: Responses collected from simulation 1.1 and simulation 1.2. They show how accounting for the values of the observable uncontrollable factors has reduced the variance of the response.

It is possible to generate a Box type model [7]. In this method the uncontrollable factors do not enter into the experiment or the modelling process explicitly.

During the experimental stage the design is replicated so that at each distinct design point there are multiple readings. The means and variances are collected at each point. They are both used as responses to be explained by the controllable factors. The  $2^{6-1}$  design was converted into a  $2^2$  design in  $x_1$  and  $x_2$  with 8 replications at each design point. The mean and variance at each design point were collected and the model found is given below.

$$\begin{aligned}\hat{\mu} &= 100.06 + 4.98x_1 + 6.45x_2 - 0.76x_1x_2 \\ \log s &= 3.04 + 0.10x_1 + 0.13x_2 - 0.07x_1x_2\end{aligned}$$

Calculus shows that the values for  $x_1$  and  $x_2$  that minimises  $\log s$  while keeping  $\hat{\mu}$  on a target of 100 is  $(-0.0026, -0.0067)$ . The model predicts a variance at that point of 433.87. A third simulation, simulation 1.3, using these  $x_i$  and the

previous  $t_i$  and  $z_i$ , gives a sampling variance of  $223.49 \pm 1.42$  and a mean of  $99.99 \pm 0.03$ .

Once again accounting for the  $t_i$  in calculating  $x_i$  has given an improvement over an accepted method.

The third approach is to calculate  $\hat{E}(y - T)^2$  for both  $\hat{x}_i^*$  and  $\hat{x}_i^{**}$  and to use the  $\hat{x}_i$  that minimises  $\hat{E}(y - T)^2$ . Using *Maple* software both expressions were derived in terms of  $t_1$  and  $t_2$  and a contour plot of  $\hat{E}(y|\hat{x}_i^*, t_i - T)^2 - \hat{E}(y|\hat{x}_i^{**}, t_i - T)^2 = 0$  is displayed in Figure 2.2. The smaller regions denote where  $\hat{x}_i^{**}$  has lower  $\hat{E}(y - T)^2$  than  $\hat{x}_i^*$ .

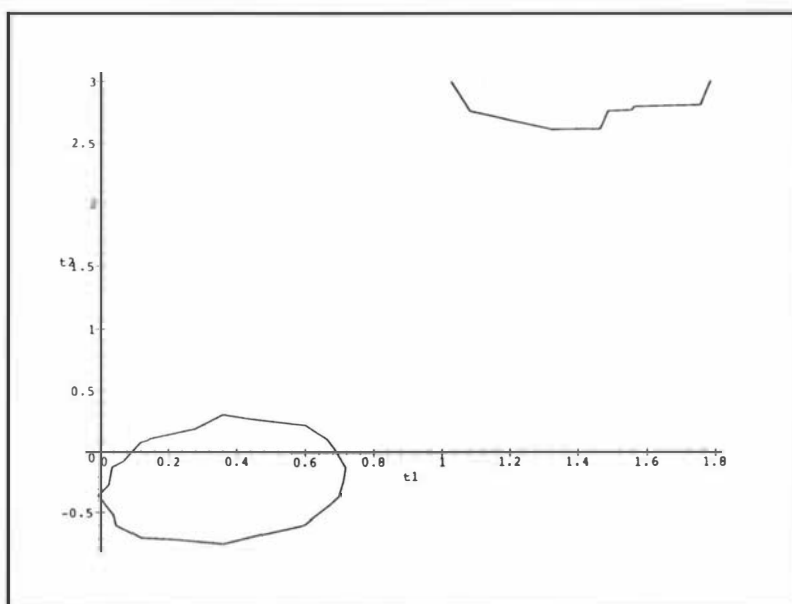


Figure 2.2: A contour plot of  $\hat{E}(y_i|\hat{x}_i^{**}, t_i - T)^2 - \hat{E}(y_i|\hat{x}_i^*, t_i - T)^2 = 0$ . Inside the two smaller regions  $\hat{x}_i^{**}$  gives the smaller  $\hat{E}(y_i|\hat{x}_i, t_i - T)^2$  value.

A further simulation, simulation 1.4, using  $\hat{x}_i^{**}$  when  $t_i$  falls in the enclosed regions and  $\hat{x}_i^*$  otherwise, gives a mean and variance for the response as  $100.18 \pm 0.03$  and  $65.44 \pm 0.62$  respectively. This approach shows little improvement over using  $\hat{x}_i^*$  exclusively because only 8.9% of the  $t_i$  fall in the enclosed region. When the probability of the  $t_i$  falling in such regions is larger this approach may give greater improvements.

A summary of the four simulations appears in Table 2.3.

Simulation	Mean	Standard Error	Variance	Standard Error
1.1	100.40	0.03	67.85	1.10
1.2	99.97	0.05	223.74	1.43
1.3	99.99	0.05	223.49	1.42
1.4	100.18	0.03	65.44	0.62

Table 2.3: A summary of the simulation results.

## 2.3 Complications

In the typical inner/outer array, only the main effects and the interactions between the controllable and uncontrollable factors are estimable, i.e. everything in equation (2.1) except  $\Sigma_3$ . However it may be the case that an experiment is designed to produce a model that contains interactions and/or quadratic terms especially when using response surface methods. Adding terms like  $\mathbf{x}_i^t \Sigma \mathbf{x}_i$  to the model gives no closed form solution for  $\mathbf{x}_i^*$ .

Others problems are that the  $\mathbf{x}_i^*$  found may not be in the experimental region or that  $\mathbf{A}$  and/or  $\mathbf{G}$  is not invertible. These problems can be overcome by solving numerically and setting appropriate upper and lower bounds on the  $\mathbf{x}_i$ .

### Further Example

Engel[13] reported an experiment to improve an injection moulding process by minimising the shrinkage of the product. The experiment consisted of seven controllable factors and three uncontrollable factors and these appear in Table 2.4.

Controllable Factors	Uncontrollable Factors
A Cycle Time	M Percentage Regrind
B Mould Temperature	N Moisture Content
C Cavity Thickness	O Ambient Temperature
D Holding Pressure	
E Injection Speed	
F Holding Time	
G Gate Size	

Table 2.4: The factors in the injection moulding experiment.

The experiment and the responses are in Table 2.5.

Cell	Controllable factors							Percentage shrinkage for the noise factors (M, N, O)			
	A	B	C	D	E	F	G	(1, 1, 1)	(1, 2, 2)	(2, 1, 2)	(2, 2, 1)
1	1	1	1	1	1	1	1	2.2	2.1	2.3	2.3
2	1	1	1	2	2	2	2	2.5	0.3	2.7	0.3
3	1	2	2	1	1	2	2	0.5	3.1	0.4	2.8
4	1	2	2	2	2	1	1	2.0	1.9	1.8	2.0
5	2	1	2	1	2	1	2	3.0	3.1	3.0	3.0
6	2	1	2	2	1	2	1	2.1	4.2	1.0	3.1
7	2	2	1	1	2	2	1	4.0	1.9	4.6	2.2
8	2	2	1	2	1	1	2	2.0	1.9	1.9	1.8

Table 2.5: The design and the responses in the injection moulding experiment.

Of the noise factors, percent regrind appears to be unmeasurable while ambient temperature is measurable. Moisture content I will deem to be measurable since it is a function of the ambient humidity and the amount of time the material is exposed to the air. All the factors appear quantitative.

The experiment was re-analyzed by Steinberg and Bursztyn[35] and they stated that perhaps two of the responses had been swapped. They were 0.3 and 2.5 in the second row, first and second column. Assuming this to be true the experiment will be analyzed again under the new method. For this example the aim will be to set shrinkage to a low target rather than minimise it.

The experiment can be written in the form give in Table 2.6.

A	B	C	D	E	F	G	M	N	O	Response
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	2.2
-1	-1	-1	1	1	1	1	-1	-1	-1	2.5
-1	1	1	-1	-1	1	1	-1	-1	-1	0.5
-1	1	1	1	1	1	-1	-1	-1	-1	2
1	-1	1	-1	1	-1	1	-1	-1	-1	3
1	-1	1	-1	-1	1	1	1	1	-1	2.1
1	1	-1	-1	1	1	-1	-1	-1	-1	4
1	1	-1	1	-1	-1	1	-1	-1	-1	2
-1	-1	-1	-1	-1	-1	-1	-1	1	1	2.1
-1	-1	-1	1	1	1	1	-1	1	1	0.3
-1	1	1	-1	-1	1	1	-1	1	1	3.1
-1	1	1	1	1	1	-1	-1	-1	1	1.9
1	-1	1	-1	1	-1	1	-1	1	1	3.1
1	-1	1	1	-1	1	-1	-1	1	1	4.2
1	1	-1	-1	1	1	-1	-1	1	1	1.9
1	1	-1	1	-1	-1	1	-1	1	1	1.9
-1	-1	-1	-1	-1	-1	-1	1	-1	1	2.3
-1	-1	-1	1	1	1	1	1	-1	1	2.7
-1	1	1	-1	-1	1	1	1	-1	1	0.4
-1	1	1	1	1	1	-1	-1	1	1	1.8
1	-1	1	-1	1	-1	1	1	-1	1	3
1	-1	1	1	-1	1	-1	1	-1	1	1
1	1	-1	-1	1	1	-1	1	-1	1	4.6
1	1	-1	1	-1	-1	1	1	-1	1	1.9
-1	-1	-1	-1	-1	-1	-1	1	1	-1	2.3
-1	-1	-1	1	1	1	1	1	1	-1	0.3
-1	1	1	-1	-1	1	1	1	1	-1	2.8
-1	1	1	1	1	1	-1	-1	1	-1	2
1	-1	1	-1	1	-1	1	1	1	-1	3
1	-1	1	1	1	-1	1	-1	1	-1	3.1
1	1	-1	-1	1	1	-1	1	1	-1	2.2
1	1	-1	1	-1	-1	1	1	1	-1	1.8

Table 2.6: The rewritten design of the injection moulding experiment.

A half normal plot of the main effects and interactions appears in Figure 2.3. The effects and interactions not falling on the line were deemed to be significant and a regression equation was derived with these. Note that the controllable factors  $B$  and  $F$  were found to be insignificant and so in production they can be set to our economic advantage.



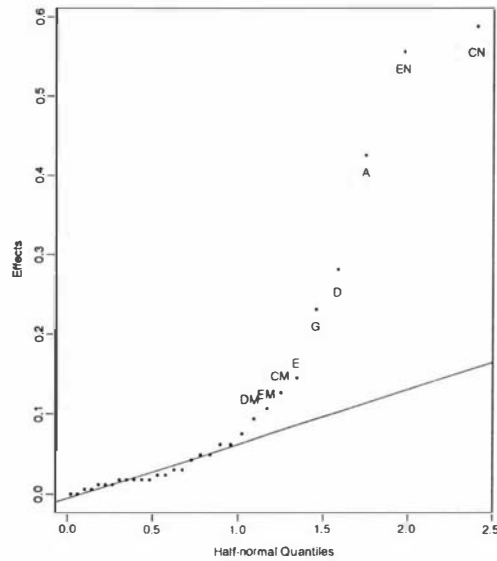


Figure 2.3: A half-normal plot of the main effects and interactions for the injection moulding experiment

The regression equation is

$$\begin{aligned} \hat{y} = & 2.25 + 0.4250A + 0.0625C - 0.2813D + 0.1438E \\ & - 0.2313G - 0.05M + 0.1063ME - 0.125MC - 0.0938MD \\ & + 0.0000N + 0.5875NC - 0.5563NE \end{aligned} \quad (2.6)$$

Therefore

$$\begin{aligned} \hat{E}(y|A, C, D, E, G, N) = & 2.25 + 0.4250A + 0.0625C - 0.2813D + 0.1437E \\ & - 0.2313G + 0.0000N + 0.5875NC - 0.55625NE \end{aligned}$$

In order to work out the variance a couple of assumptions have to be made. I will assume that the distributions of  $M$ , % regrind, and  $N$ , the ambient temperature are Normal and centered about 0. Secondly I will assume that 95% of the distribution lies between their levels i.e. they have standard deviations of  $\frac{1}{2}$ .

Therefore

$$\widehat{Var}(y|A, C, D, E, G, N) = var(M)(-0.05 - 0.125C + 0.1063E - 0.0938D)^2 + \hat{\sigma}_\epsilon^2$$

In this case the target will be set at 1.6% shrinkage. A quadratic program was set up using the *student ampl* software package with  $\widehat{Var}(y|A, C, D, E, G, N)$  as the

objective that is to be minimised subject to  $\hat{E}(y|A, C, D, E, G, N) = 1.6$ . Also since the experiment was done in the unit hypercube all controllable factors will be constrained to lie between  $\pm 1$ . During production  $N$  can vary and the best approach would be to solve the quadratic program for the particular value for  $N$  at that time. This case will be simulated later. Generally this is likely to be expensive and approximations to the solution can be found. In the approximate method the idea will be to observe the solution to the quadratic program for different values of  $N$  and relate the solutions for the controllable factors to  $N$  by regression equations. Technically only values of  $N$  between  $\pm 1$  should be considered, however in practise values outside this range (and in this case 5 out of 100) will be seen, so  $N$  was observed over the range  $\pm 3$ . For each value of  $N$  a solution from the quadratic program was found for each of the controllable factors. These were plotted against  $N$  in Figure 2.4.

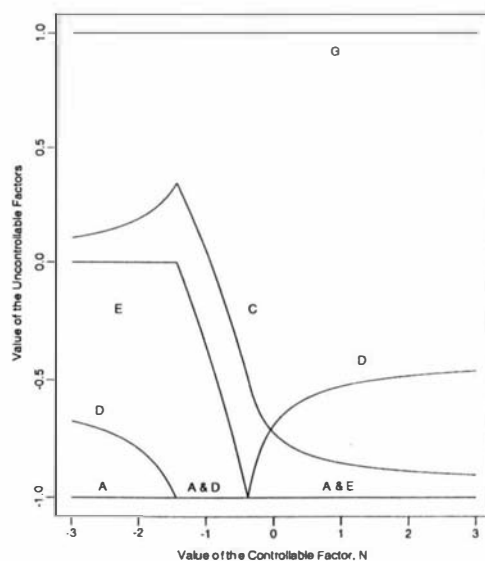


Figure 2.4: A plot of the observable uncontrollable factor,  $N$  versus the controllable factors,  $A, C, D, E, G$ .

From this it can be seen that factor  $A$  should be always set at  $-1$  and  $G$  at  $1$ . The others change their values along smooth curves with breakpoints at  $-1.4425$  and  $-0.3825$ . Regression equations were fitted to these smooth curves in order that the values for factors  $C, D$  and  $E$  could be predicted given a value of  $N$ . In this case we want these equations to be good predictors so that parsimony is not

necessary. Terms were added to each equation until a term became insignificant or the residual squared error reached some threshold. In this case the threshold was 1e-5. This threshold couldn't be reached in the regression for equation *C* in the range  $(-0.3825, 3)$  and this equation does not predict as well as the others. The solution is in Table 2.7.

<i>A</i>	= -1	$-3 < N < 3$	
<i>B</i>	our choice	$-3 < N < 3$	
<i>C</i>	= {	$19.31 + 59.20N + 83.18N^2 + \dots + 0.015N^8$	$-3 < N \leq -1.4425$
		$-0.93 - 1.25N - 0.34N^2 - 0.07N^3 - 0.01N^4$	$-1.4425 < N \leq -0.3825$
		$-0.72 - 0.29N + 0.39N^2 + \dots + 0.0002N^{14}$	$-0.3825 < N < 3$
<i>D</i>	= {	$-26.28 - 78.90N - 110.85N^2 + \dots - 0.02N^8$	$-3 < N \leq -1.4425$
		-1	$-1.4425 < N \leq -0.3825$
		$-0.70 + 0.39N - 0.52N^2 + \dots + 0.0003N^{13}$	$-0.3825 < N < 3$
<i>E</i>	= {	0	$-3 < N \leq -1.4425$
		$-1.51 - 1.47N - 0.40N^2 - 0.09N^3 - 0.01N^4$	$-1.4425 < N \leq -0.3825$
		-1	$-0.3825 < N < 3$
<i>F</i>	our choice	$-3 < N < 3$	
<i>G</i>	= 1	$-3 < N < 3$	

Table 2.7: The solution giving the relationship between the controllable factors *A, B, C, D, E, F, G* and the observable uncontrollable factor, *N*, that minimises the variance and keeps the mean on target.

Since the underlying model is not known we cannot tell how the solution would do. This could only be found by using the solution practically. However the two different methods can be compared to see if there is any difference between them. Two simulations were done with the same 100,000 random variates for each of *N* and *M*. In the first simulation for each value of *N* the quadratic program was solved to give values for the controllable factors. These were used in equation 2.6 to generate the responses  $\hat{y}_1$ . The variance of  $\hat{y}_1$  was 2.26e-12. In the second simulation the values of the controllable factors were chosen using the solution in Table 2.7 and these were used in equation 2.6 to generate the responses  $\hat{y}_2$ . The variance of  $\hat{y}_2$  was 9.97e-11. The two methods show little difference in the

response variance. With any reasonable error variance these response variances would be swamped and the two methods would be indistinguishable in outcome. The responses for the second simulation were plotted against  $N$  in Figure 2.5.

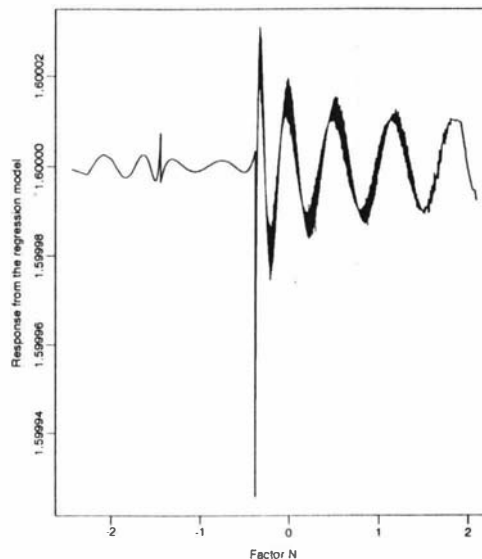


Figure 2.5: A plot of the response to the regression equation using the solutions in Table 2.7 for the factors and 100,000 random variates for  $M$  and  $N$ .

After  $N = -0.3825$  the variance in the response is greater. This is due to the poorer predictive power of the regression model for factor  $C$  in this region.

## 2.4 Implicit Introduction of Uncontrollable Factors

The previous sections have dealt with the case where the uncontrollable factors have been explicitly introduced into the experiment and this was demonstrated by using Response Surface methods. There are methods that use the implicit introduction of uncontrollable factors. Instead of modelling the quality response as in the RSM method or the Split Plot method, these methods produce a joint model for the mean and for the variance. These include

- the transformation and data analytic approach
- generalised linear models approach

- restricted generalised models approach

These methods can be extended very easily to include the idea of separating the uncontrollable factors into those that can be observed and into those that are unobservable. This extra information from the observable uncontrollable factors can be used to our advantage.

## The Design Stage

In these three methods given above the uncontrollable factors need not be explicitly part of the experimental design. They are introduced into the experiment by having replications at each of the design points. The assumption is that the uncontrollable factors follow their normal pattern of behaviour. If this assumption is likely to be invalid then these factors should be explicitly introduced into the experiment i.e. given factor names and levels and with the experiment designed accordingly.

In this new method the unobservable uncontrollable factors are allowed to enter into the experiment implicitly while the observable uncontrollable factors enter into the experiment explicitly.

## The Modelling Stage

The methods listed above produce a joint model for the mean and variance based on the controllable factors. Recall that these methods called for separation of the mean and variance of the quality response i.e. there was no relationship between the mean and variance. The transformation method used a transformation of the quality response to effect this separation. The variances of this transformed data set are used as the responses for the variance model.

For the other two methods the variance was separated into two parts  $V(\mu)$  and  $\phi$ . The former being that part of the variance that was related to the mean and the latter being that part of the variance unrelated to the mean. In these cases  $\phi$  was used as the response for the variance model.

In this new method, at the modelling stage, the controllable and observable uncontrollable factors are treated similarly. A joint model for the mean and variance is put together using the controllable and observable uncontrollable factors.

The final stage is to find the values of the controllable factors by minimising the variance while keeping the mean model on target. This may be possible to solve using Lagrangian Minimisation techniques. However it is most likely that linear/quadratic programming methods are needed to force any solutions for the controllable factors to be within the region on the experimental design.

### 2.4.1 Example

In Section 1.9.5 the Generalised Linear Models method was demonstrated and the following model was found. The model is

$$\begin{aligned} \sqrt{\hat{\mu}} &= 12.46 + 0.97 \times A + 1.49 \times D \\ \log \hat{\phi} &= -0.59 + 0.33 \times A + 0.34 \times C \\ \text{and } Var(y_i) &= \phi_i V(\mu_i) \\ &= \exp[-0.59 + 0.33A + 0.34C] \times (12.46 + 0.97A + 1.49D)^2 \end{aligned}$$

Suppose  $A$  was an observable uncontrollable factor and that the levels of -1 and 1 were chosen so that 95% of the values of  $A$  were known to fall between them. Minimising the variance while keeping the mean on target of 144 or  $\sqrt{\hat{\mu}} = 12$  would involve minimising

$$-0.59 + 0.33A + 0.34C \text{ subject to } 12 = 12.46 + 0.97A + 1.49D$$

The only solution to the latter equation is having

$$\hat{D} = (12 - 12.46 - 0.97A)/1.49 \quad A = -1 \dots 1$$

Minimising the variance would involve setting  $C$  so that  $-0.59 + 0.33A + 0.34C$  is a minimum. That is when  $\hat{C}$  is at  $-1$ , the lowest value available to  $C$  within the experimental region.

If the target was 100 then

$$\hat{D} = (10 - 12.64 - 0.97A)/1.49 \quad A = -1 \dots 1$$

However the equation for  $\hat{D}$  would give values of  $D$  that were not in the range  $(-1, 1]$  but in the range  $[-1, -2.3]$ . Since the model loses validity outside the experimental region it would be advisable to do more experimentation. Using smaller levels of  $D$ , say -1 and -2.3, should be considered. Also since the variance is decreasing as  $C$  decreases, smaller values of  $C$  maybe worth investigating as well. However there may be practical limitations on how low  $C$  can go.

## 2.5 Allowing For Bias

The goal has been to choose values for the controllable factors  $\mathbf{x}_i$  that keep the mean of the quality control variable,  $Y$ , on target while minimising its variance. If the constraint of keeping the mean on target is relaxed then lower values for the variance are possible. However it is still important to keep the mean of  $Y$  as close to the target as possible. The Mean Square Error about the target gives an indication of how far we expect the quality control variable to be from the target. It is

$$E_Y((Y|\mathbf{x}_i, \mathbf{t}_i - T)^2) = \text{Var}_Y(y_i|\mathbf{x}_i, \mathbf{t}_i) + (E_Y(y_i|\mathbf{x}_i, \mathbf{t}_i) - T)^2$$

It is both a function of the variance and a function of the difference between the mean and the target.

For model one of chapter two the MSE can be represented as

$$\begin{aligned} E_Y((Y|\mathbf{x}_i, \mathbf{t}_i - T)^2) &= \text{Var}_Y(Y|\mathbf{x}_i, \mathbf{t}_i) + (E_Y(Y|\mathbf{x}_i, \mathbf{t}_i) - T)^2 \\ &= \sigma_\epsilon^2 + (\beta_3 + \Sigma_2 \mathbf{x}_i + \Sigma_3 \mathbf{t}_i)^t \Lambda_2 (\beta_3 + \Sigma_2 \mathbf{x}_i + \Sigma_3 \mathbf{t}_i) \\ &\quad + (\beta_0 + \mathbf{x}_i^t \beta_1 + \mathbf{t}_i^t \beta_2 + \mathbf{x}_i^t \Sigma_1 \mathbf{t}_i - T)^2 \end{aligned}$$

The following theorem finds a value for  $\mathbf{x}_i$  that minimises the MSE.

**Theorem 2.10** *If*

- 1) *the inverse exists*
- 2)  $\Lambda_2$  *is positive definite matrix and*
- 3)  $\Sigma_2$  *is of rank*  $r$

*then the minimum of*  $E((Y|x_i, t_i - T)^2)$  *with respect to*  $\mathbf{x}_i$  *is*

$$\begin{aligned} \mathbf{x}_i^{***} &= -(\Sigma_2^t \Lambda_2 \Sigma_2 + (\beta_1 + \Sigma_1 \mathbf{t}_i)(\beta_1^t + \mathbf{t}_i^t \Sigma_1^t))^{-1} \\ &\quad \times (\Sigma_2^t \Lambda_2 \beta_3 + \Sigma_2^t \Lambda_2 \Sigma_3 \mathbf{t}_i + (\beta_1 + \Sigma_1 \mathbf{t}_i)(\beta_0 + \mathbf{t}_i^t \beta_2 - T)) \end{aligned}$$

**Proof**

Now

$$\begin{aligned} E((y_i|x_i, t_i - T)^2) &= \sigma_\epsilon^2 + (\beta_3 + \Sigma_2 \mathbf{x}_i + \Sigma_3 \mathbf{t}_i)^t \Lambda_2 (\beta_3 + \Sigma_2 \mathbf{x}_i + \Sigma_3 \mathbf{t}_i) \\ &\quad + (\beta_0 + \mathbf{x}_i^t \beta_1 + \mathbf{t}_i^t \beta_2 + \mathbf{x}_i^t \Sigma_1 \mathbf{t}_i - T)^2 \\ &= \sigma_\epsilon^2 + (\beta_3 + \Sigma_2 \mathbf{x}_i + \Sigma_3 \mathbf{t}_i)^t \Lambda_2 (\beta_3 + \Sigma_2 \mathbf{x}_i + \Sigma_3 \mathbf{t}_i) \\ &\quad + (\beta_0 + \beta_1^t \mathbf{x}_i + \mathbf{t}_i^t \beta_2 + \mathbf{t}_i^t \Sigma_1^t \mathbf{x}_i - T)^2 \end{aligned}$$

Differentiating this with respect to  $\mathbf{x}_i$  gives

$$\begin{aligned} \frac{\partial E((y_i|\mathbf{x}_i, t_i - T)^2)}{\partial \mathbf{x}_i} &= 2\Sigma_2^t \Lambda_2 \beta_3 + 2\Sigma_2^t \Lambda_2 \Sigma_3 t_i + 2\Sigma_2^t \Lambda_2 \Sigma_2 \mathbf{x}_i \\ &\quad + 2(\beta_1 + \Sigma_1 t_i)(\beta_0 + \beta_1^t \mathbf{x}_i + t_i^t \beta_2 + t_i^t \Sigma_1^t \mathbf{x}_i - T) \\ &= 2(\Sigma_2^t \Lambda_2 \Sigma_2 + (\beta_1 + \Sigma_1 t_i)(\beta_1^t + t_i^t \Sigma_1^t)) \mathbf{x}_i + 2\Sigma_2^t \Lambda_2 \beta_3 \\ &\quad + 2\Sigma_2^t \Lambda_2 \Sigma_3 t_i + 2(\beta_1 + \Sigma_1 t_i)(\beta_0 + t_i^t \beta_2 - T) \end{aligned}$$

Solving this for  $x_i$  gives

$$\begin{aligned} \mathbf{x}_i^{***} &= -(\Sigma_2^t \Lambda_2 \Sigma_2 + (\beta_1 + \Sigma_1 t_i)(\beta_1^t + t_i^t \Sigma_1^t))^{-1} \\ &\quad \times (\Sigma_2^t \Lambda_2 \beta_3 + \Sigma_2^t \Lambda_2 \Sigma_3 t_i + (\beta_1 + \Sigma_1 t_i)(\beta_0 + t_i^t \beta_2 - T)) \end{aligned}$$

The second derivative with respect to  $\mathbf{x}_i$  is

$$\frac{\partial^2 E((y_i|\mathbf{x}_i, t_i - T)^2)}{\partial \mathbf{x}_i \partial \mathbf{x}_i^t} = 2(\Sigma_2^t \Lambda_2 \Sigma_2 + (\beta_1 + \Sigma_1 t_i)(\beta_1^t + t_i^t \Sigma_1^t))$$

If  $\frac{\partial^2 E((y_i|\mathbf{x}_i, t_i - T)^2)}{\partial \mathbf{x}_i \partial \mathbf{x}_i^t}$  is positive definite then  $\mathbf{x}_i^{***}$  is the minimum critical point. If we take that  $\Lambda_2$  is positive definite then

$$\Sigma_2^t \Lambda_2 \Sigma_2$$

will be positive definite if  $\Sigma_2$  is of rank  $r$ .

Let

$$\mathbf{R} = (\beta_1 + \Sigma_1 t_i)$$

and  $\mathbf{0}_{(p-1) \times p}$  be a  $(p-1) \times p$  matrix of zeros. Combine these to give  $\mathbf{S}$  where

$$\mathbf{S} = \begin{pmatrix} \mathbf{R}^t \\ \mathbf{0}_{(p-1) \times p} \end{pmatrix}$$

a  $p \times p$  matrix. Now

$$\mathbf{R}\mathbf{R}^t = (\beta_1 + \Sigma_1 t_i)(\beta_1^t + t_i^t \Sigma_1^t) = \mathbf{S}^t \mathbf{S}$$

Since  $\mathbf{S}$  is square and has the same rank as  $\mathbf{R}^t$  then  $\mathbf{R}\mathbf{R}^t$  is positive semi-definite. This means

$$\frac{\partial^2 E((y_i|\mathbf{x}_i, t_i - T)^2)}{\partial \mathbf{x}_i \partial \mathbf{x}_i^t} = 2(\Sigma_2^t \Lambda_2 \Sigma_2 + (\beta_1 + \Sigma_1 t_i)(\beta_1^t + t_i^t \Sigma_1^t))$$

is positive definite. ■



In theory if  $\frac{\partial^2 E((y_i | \mathbf{x}_i, t_i - T)^2)}{\partial \mathbf{x}_i \partial \mathbf{x}_i^t}$  is positive definite then it will be non-singular however in practice if

$$(\beta_1 + \Sigma_1 t_i)(\beta_1^t + t_i^t \Sigma_1^t)$$

is to ‘large’ then it will swamp

$$\Sigma_2^t \Lambda_2 \Sigma_2$$

Trying to find the inverse by computer will fail as the computer tries to invert what appears to be a singular matrix.

In practice, the parameters will be replaced with their estimates.

### 2.5.1 Simulation Example Continued

In this simulation  $\hat{\mathbf{x}}_i^*$  and  $\hat{\mathbf{x}}_i^{***}$  are to going to be compared using the data from Section 2.2. In the previous example, the conditions for  $\hat{\mathbf{x}}_i^*$  to be the minimum of the variance when the mean is on target, were met. If  $\Lambda_2$  is positive definite and  $\Sigma_2$  is of rank  $r$  then  $\hat{\mathbf{x}}_i^{***}$  is the minimum of the MSE. Now recall that  $\Lambda_2$  was the identity matrix so it is positive definite. Also

$$\hat{\Sigma}_2 = \begin{pmatrix} 6.11 & 0 \\ -6.98 & -7.36 \end{pmatrix}$$

is of rank 2 and  $r = 2$ . And so the conditions are met.

Twenty-one thousand values for  $t_1$  and  $t_2$ , and  $z_1$  and  $z_2$  were generated in sets of 1000 with the same distributions as before. Two different versions of  $\mathbf{x}_i$  were generated,  $\hat{\mathbf{x}}_i^*$  and  $\hat{\mathbf{x}}_i^{***}$ . The function of  $\hat{\mathbf{x}}_i^*$  in terms of  $t_i$  was given in table 2.2. The function of  $\hat{\mathbf{x}}_i^{***}$  in terms of  $t_i$  is given in Table 2.8.

$$(x1, x2)^t = \begin{pmatrix} \frac{2476 - 6188t_1 - 277t_2 - 685t_1^2 + 1445t_1 t_2 + 489t_2^2 + 252t_1 t_2^2 - 277t_2^3 + 1e - 8t_1^2 t_2^2 + 2e - 8t_2^3 t_1}{3643 - 1288t_2 + 398t_2^2 + 442t_1 - 789t_1 t_2 + 690t_1^2 + 1e - 8t_1^2 t_2^2} \\ \frac{-4091 + 6627t_1 - 3300t_2 - 1151t_1^2 - 879t_1 t_2 + 52t_2^2 - 418t_1^2 t_2 + 460t_1 t_2^2 - 1e - 8t_1^3 t_2}{3643 - 1288t_2 + 398t_2^2 + 442t_1 - 789t_1 t_2 + 690t_1^2 + 1e - 8t_1^2 t_2^2} \end{pmatrix}$$

Table 2.8: The equation giving the relationship between  $\hat{\mathbf{x}}_i^{***}$  and  $t_i$  found using maple.

These values were used to generate  $Y$ , the quality variable, according to the underlying model given in equation 2.5.

A histogram comparing the two resulting distributions for the first 10,000 values of  $Y$  appears in Figure 2.6.

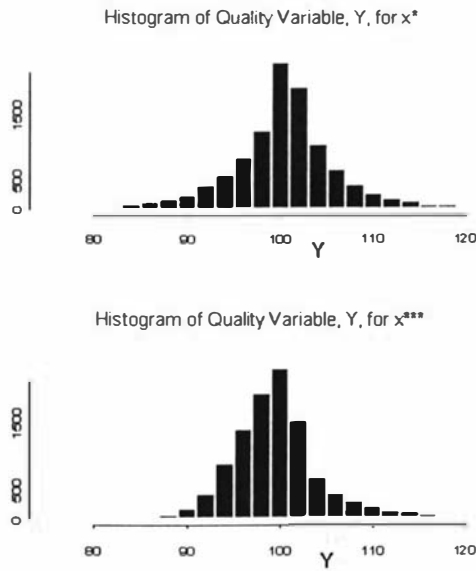


Figure 2.6: Responses collected in histograms for  $Y$  when  $T=100$  for  $\hat{x}_i^*$  and  $\hat{x}_i^{***}$  respectively.

The histogram for simulation 2.1, using  $\hat{x}_i^*$  has a bell-shape but is more spread then simulation 2.2 which uses  $\hat{x}_i^{***}$ . The histogram for simulation 2.2 does not look bell-shaped. To the left of the target value it has the shape of a triangular distribution peaking at the target. To the right of the target it has the shape of an exponential distribution.

A comparison of the results for these two simulations appears in Table 2.9. The first simulation are the results for  $Y$  when using  $\hat{x}_i^*$ . The second simulations are the results  $Y$  when using  $\hat{x}_i^{***}$ .

The means and variances were collected from each of the 21 sets of 1000 responses and were averaged to give the mean and variance reported in the table. The standard error for each reported mean and variance was derived from the sample variance of the 21 means and 21 variances respectively.

Simulation	Mean	Standard Error	Variance	Standard Error
2.1	100.36	0.04	66.92	1.07
2.2	100.05	0.03	47.46	0.86

Table 2.9: Responses collected in histograms for  $Y$  when  $T=100$  for  $\hat{x}_i^*$  and  $\hat{x}_i^{***}$  respectively.

From the table it can be seen that the mean of  $Y$  is closer to the target when using  $\hat{x}_i^{***}$ . The mean of  $Y$  for  $\hat{x}_i^{***}$  is within two standard errors of the target, while the mean for  $x_i^*$  is not.

The variance of  $Y$  is much smaller using  $\hat{x}_i^{***}$  compared with using  $\hat{x}_i^*$ .

The simulations were run again but with a target value of 125 rather than 100. The versions  $\hat{x}_i^*$  and  $\hat{x}_i^{***}$  were generated once again and used in the underlying model of equation 2.5 to generate  $Y$ . A histogram of the first 10,000  $Y$  values for each version of  $x_i$  appears in Figure 2.7.

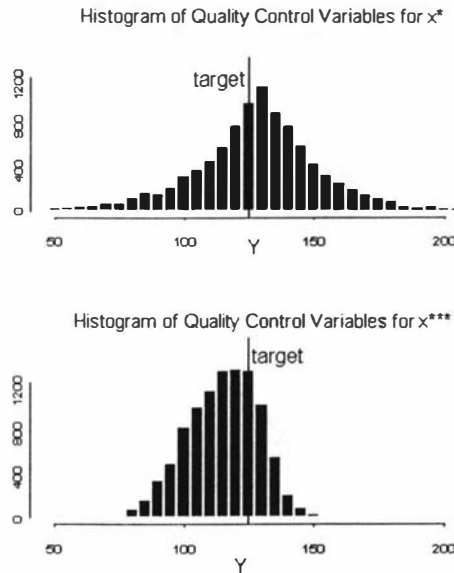


Figure 2.7: Responses collected in histograms for  $Y$  when  $T=125$  for  $\hat{x}_i^*$  and  $\hat{x}_i^{***}$  respectively.

The first histogram which records  $Y$ , generated from  $\hat{x}_i^*$ , has a mean that is much closer to the target value of 125 but is much more spread out. The second histogram which records  $Y$  generated from  $\hat{x}_i^{***}$  has a mean for  $Y$  which is about 10 units from the target of 125 but is much more compact about its mean than for the first simulation.

The simulation results for twenty-one thousand values is in Table 2.10. Since  $\hat{x}_i^{***}$  gave a mean for  $Y$  that was far from the target the Mean Square Error for both variables appears in the table as well.

Simulation	Mean	Std. Error	Variance	Std Error	MSE	Std. Error
3.1	128.21	0.23	872.20	19.89	873.52	20.76
3.2	114.69	0.08	213.37	2.10	309.68	3.36

Table 2.10: A summary of the simulation results of the mean, variance and MSE(T=125) for Y.

In the general case it would seem that  $\hat{x}_i^{***}$  would be recommended over  $\hat{x}_i^*$  as it is not too far from the target and has lower variance. However it may be needed to get the mean closer to the target. One approach could be to try and exert some control over  $t_i$ .

Lin and Yu [18] gave another approach. They recommend using an adjusted version of the Mean Square Error. They introduced two new variables,  $\lambda_1 > 0$  and  $\lambda_2 > 0$ , to give the adjusted MSE

$$MSE^*(T) = \lambda_1 Var(Y) + \lambda_2 (E(Y) - T)^2$$

The values of  $\lambda_1$  and  $\lambda_2$  are adjusted to suit the preference of the experimenter. When  $\lambda_1 = \lambda_2 = 1$  then the adjusted MSE is equivalent to the MSE. When  $\lambda_1 = 1$  and  $\lambda_2$  tends towards infinity then minimising the adjusted MSE is equivalent to minimising the variance of Y subject to the mean being on target.

The value that  $x_i$  takes will be adjusted for the values of  $\lambda_1$  and  $\lambda_2$ .

**Theorem 2.11** *If*

- 1) *the inverse exists,*
  - 2)  $\Lambda_2$  *is positive definite*
  - 3)  $\Sigma_2$  *is of rank r and*
  - 4)  $\lambda_1 > 0$  *and*  $\lambda_2 > 0$
- then the minimum of*

$$MSE^*(T) = \lambda_1 Var(Y) + \lambda_2 (E(Y) - T)^2$$

*with respect to*  $x_i$  *is*

$$\begin{aligned} x_i^{****} &= -(\lambda_1 \Sigma_2^t \Lambda_2 \Sigma_2 + \lambda_2 (\beta_1 + \Sigma_1 t_i) (\beta_1^t + t_i^t \Sigma_1^t))^{-1} \\ &\quad \times (\lambda_1 \Sigma_2^t \Lambda_2 \beta_3 + \lambda_1 \Sigma_2^t \Lambda_2 \Sigma_3 t_i + \lambda_2 (\beta_1 + \Sigma_1 t_i) (\beta_0 + t_i^t \beta_2 - T)) \end{aligned}$$

**Proof**

Now

$$\begin{aligned} MSE^*(T) &= \lambda_1 Var(Y) + \lambda_2 (E(Y) - T)^2 \\ &= \lambda_1 \sigma_\epsilon^2 + \lambda_1 (\beta_3 + \Sigma_2 \mathbf{x}_i + \Sigma_3 \mathbf{t}_i)^t \Lambda_2 (\beta_3 + \Sigma_2 \mathbf{x}_i + \Sigma_3 \mathbf{t}_i) \\ &\quad + \lambda_2 (\beta_0 + \beta_1^t \mathbf{x}_i + \mathbf{t}_i^t \beta_2 + \mathbf{t}_i^t \Sigma_1^t \mathbf{x}_i - T)^2 \end{aligned}$$

Differentiating this with respect to  $\mathbf{x}_i$  gives

$$\begin{aligned} \frac{\partial E((y_i | \mathbf{x}_i, \mathbf{t}_i - T)^2)}{\partial \mathbf{x}_i} &= \lambda_1 (2 \Sigma_2^t \Lambda_2 \beta_3 + 2 \Sigma_2^t \Lambda_2 \Sigma_3 \mathbf{t}_i + 2 \Sigma_2^t \Lambda_2 \Sigma_2 \mathbf{x}_i) \\ &\quad + 2 \lambda_2 (\beta_1 + \Sigma_1 \mathbf{t}_i) (\beta_0 + \beta_1^t \mathbf{x}_i + \mathbf{t}_i^t \beta_2 + \mathbf{t}_i^t \Sigma_1^t \mathbf{x}_i - T) \\ &= 2 (\lambda_1 \Sigma_2^t \Lambda_2 \Sigma_2 + \lambda_2 (\beta_1 + \Sigma_1 \mathbf{t}_i) (\beta_1^t + \lambda_1 \mathbf{t}_i^t \Sigma_1^t)) \mathbf{x}_i + 2 \lambda_1 \Sigma_2^t \Lambda_2 \beta_3 \\ &\quad + 2 \lambda_1 \Sigma_2^t \Lambda_2 \Sigma_3 \mathbf{t}_i + 2 \lambda_2 (\beta_1 + \Sigma_1 \mathbf{t}_i) (\beta_0 + \mathbf{t}_i^t \beta_2 - T) \end{aligned}$$

Solving this for  $\mathbf{x}_i$  gives

$$\begin{aligned} \mathbf{x}_i^{****} &= -(\lambda_1 \Sigma_2^t \Lambda_2 \Sigma_2 + \lambda_2 (\beta_1 + \Sigma_1 \mathbf{t}_i) (\beta_1^t + \mathbf{t}_i^t \Sigma_1^t))^{-1} \\ &\quad \times (\lambda_1 \Sigma_2^t \Lambda_2 \beta_3 + \lambda_1 \Sigma_2^t \Lambda_2 \Sigma_3 \mathbf{t}_i + \lambda_2 (\beta_1 + \Sigma_1 \mathbf{t}_i) (\beta_0 + \mathbf{t}_i^t \beta_2 - T)) \end{aligned}$$

The second derivative with respect to  $\mathbf{x}_i$  is

$$\frac{\partial^2 E((y_i | \mathbf{x}_i, \mathbf{t}_i - T)^2)}{\partial \mathbf{x}_i \partial \mathbf{x}_i^t} = 2 (\lambda_1 \Sigma_2^t \Lambda_2 \Sigma_2 + \lambda_2 (\beta_1 + \Sigma_1 \mathbf{t}_i) (\beta_1^t + \mathbf{t}_i^t \Sigma_1^t))^t$$

If  $\frac{\partial^2 E((y_i | \mathbf{x}_i, \mathbf{t}_i - T)^2)}{\partial \mathbf{x}_i \partial \mathbf{x}_i^t}$  is positive definite then  $\mathbf{x}_i^{****}$  will be the minimum critical point. If we take that  $\Lambda_2$  is positive definite then  $\lambda_1 \Sigma_2^t \Lambda_2 \Sigma_2$  will be positive definite if  $\Sigma_2$  is of rank  $r$  and  $\lambda_1 > 0$ .

Let

$$\mathbf{R} = \lambda_2^{\frac{1}{2}} (\beta_1 + \Sigma_1 \mathbf{t}_i)$$

and  $\mathbf{0}_{(p-1) \times p}$  be a  $(p-1) \times p$  matrix of zeros. Combine these to give  $\mathbf{S}$  where

$$\mathbf{S} = \lambda_2^{\frac{1}{2}} \begin{pmatrix} \mathbf{R}^t \\ \mathbf{0}_{(p-1) \times p} \end{pmatrix}$$

a  $p \times p$  matrix.

Now

$$\mathbf{R} \mathbf{R}^t = \lambda_2 (\beta_1 + \Sigma_1 \mathbf{t}_i) (\beta_1^t + \mathbf{t}_i^t \Sigma_1^t) = \mathbf{S}^t \mathbf{S}$$

Since  $S$  is square and has the same rank as  $R^t$  then  $RR^t$  is positive semi-definite. This means

$$\frac{\partial^2 E((y_i|\mathbf{x}_i, t_i - T)^2)}{\partial \mathbf{x}_i \partial \mathbf{x}_i^t}$$

is positive definite. ■

In theory if  $\frac{\partial^2 E((y_i|\mathbf{x}_i, t_i - T)^2)}{\partial \mathbf{x}_i \partial \mathbf{x}_i^t}$  is positive definite then it will be non-singular. In practice if  $(\lambda_2(\beta_1 + \Sigma_1 t_i)(\beta_1^t + t_i^t \Sigma_1^t))$  or  $\lambda_2$  is to 'large' or  $\lambda_1$  to 'small' then  $(\lambda_2(\beta_1 + \Sigma_1 t_i)(\beta_1^t + t_i^t \Sigma_1^t))$  will swamp  $\lambda_1 \Sigma_2^t \Lambda_2 \Sigma_2$ . Trying to find the inverse by computer will fail as the computer tries to invert what appears to it to be a singular matrix.

### 2.5.2 Simulation Example Continued

In this simulation different versions of  $\hat{\mathbf{x}}_i^{***}$  are to going to be compared using the data from Section 2.2. If  $\Lambda_2$  is positive definite and  $\Sigma_2$  is of rank  $r$  and  $\lambda_1$  and  $\lambda_2$  are both greater than zero then  $\hat{\mathbf{x}}_i^{***}$  is the minimum of the MSE. Now recall that  $\Lambda_2$  was the identity matrix so it is positive definite. Also

$$\hat{\Sigma}_2 = \begin{pmatrix} 6.11 & 0 \\ -6.98 & -7.36 \end{pmatrix}$$

is of rank 2 and  $r = 2$ . And so as long as  $\lambda_1$  and  $\lambda_2$  are chosen to be both greater then the conditions are met.

Ten thousand values for  $t_1$  and  $t_2$ , and  $z_1$  and  $z_2$  were generated in sets of 1000 with the same distribution as before. Four different versions of  $\mathbf{x}_i$  were generated. They were  $\hat{\mathbf{x}}_i^{****}$  with different values for  $\lambda_1$  and  $\lambda_2$ .

These values were used to generate  $Y$ , the quality variable, according to the underlying model given in equation 2.5. The simulations were run with a target value of 125.

A histogram of the first 10,000  $Y$  values for  $\hat{\mathbf{x}}_i^{****}$  with  $\lambda_2=1, 2, 5, \text{ and } 10$ , appears in Figure 2.8.

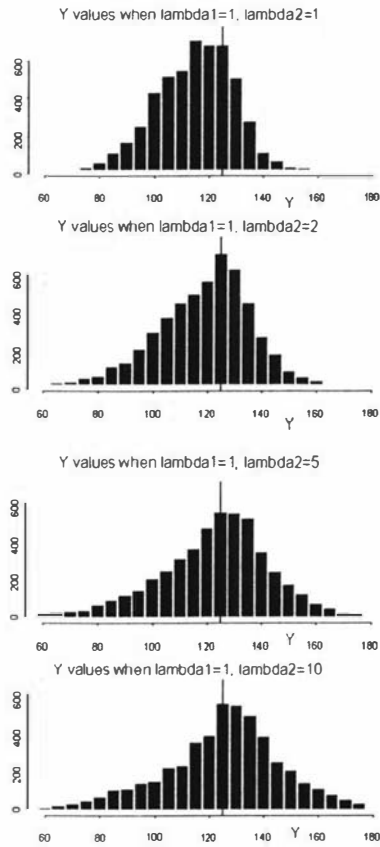


Figure 2.8: Responses collected in histograms for  $Y$  when  $T=125$  for  $\hat{\mathbf{x}}_i^{***}$  when  $\lambda_1 = 1$  and  $\lambda_2 = 1, 2, 5$  and  $10$  respectively.

As  $\lambda_2$  increases the mean of  $Y$  moves closer to the target. However the variance and Mean Square Error for  $Y$  increases.

The simulation results for ten thousand values of the quality response are in Table 2.11. Since  $\hat{\mathbf{x}}_i^{***}$  gave a mean for  $Y$  that was different from the target the Mean Square Error appears in the table as well. Also included in the table are the results from simulation 3 from Table 2.10.

Simulation	$\lambda_2$	Mean	Std. Error	Variance	Std Error	MSE	Std. Error
3.2	1	114.69	0.08	213.37	2.10	309.68	3.36
4.1	2	118.86	0.15	316.08	3.66	353.74	4.64
4.2	5	123.10	0.26	489.35	9.06	493.52	9.66
4.3	10	124.94	0.24	615.36	13.71	615.84	13.65
4.4	200	127.47	0.30	839.12	25.65	845.21	25.65
3.1	NA	128.21	0.23	872.20	19.89	873.52	20.76

Table 2.11: A summary of the simulation results of the mean, variance and MSE( $T=125$ ) for  $Y$  for different values of  $\lambda_2$

The last line in the table is the simulation results when using  $\hat{x}_i^*$  i.e. keeping the mean of  $Y$  on target while minimising the variance. This is equivalent to having  $\lambda_2 = \infty$ . The mean of  $Y$  is not on target since  $\hat{x}_i^*$  is being used to estimate  $x_i^*$ . In using the estimate a degree of error has been introduced.

The first line of the table contains the results when using  $\hat{x}_i^{**}$ . These are found by minimising the MSE about 125. These results show the most bias but the least variance and MSE. The other lines of the table are for  $\lambda_2 = 2, 5, 10$  and 200. As  $\lambda_2$  increases then the variance and MSE of  $Y$  increases. Also the means are increasing to the value of the mean for simulation 3.1.

Using  $\lambda_1$  and  $\lambda_2$  to adjust the Mean Square Error can be beneficial in getting the mean of the quality control variable closer to the target but at the expense of its variance. Choosing values of  $\lambda_1$  and  $\lambda_2$  is a subjective decision that balances minimising the bias and minimising the variance.

## 2.6 Bias and the Implicit Introduction of Uncontrollable Factors

In the previous section models based on the quality response were used to find a setting of the controllable factors that minimised the Mean Square Error of the quality response about the target. Models that are based on the mean and variance of the quality response can be used in a similar way. These models came about from experimental designs where the uncontrollable, unobservable factors need not be introduced into the designs explicitly.



### Example Continued

In section 1.9.5 the following model based on the mean and variance of the quality response was found. In section 2.4.1 the target was given as  $T = 144$ . The model is

$$\begin{aligned}\sqrt{\hat{\mu}} &= 12.46 + 0.97 \times A + 1.49 \times D \\ \log \hat{\phi} &= -0.59 + 0.33 \times A + 0.34 \times C \\ \text{and } Var(y_i) &= \phi_i V(\mu_i) \\ &= \exp[-0.59 + 0.33A + 0.34C] \times (12.46 + 0.97A + 1.49D)^2\end{aligned}$$

Suppose that  $A$  is an observable uncontrollable factor and that the levels of -1 and 1, used in the experiment, were chosen so that 95% of the values of  $A$  were known to fall between them. Also take that the remaining factors,  $C$  and  $D$ , are controllable factors. In the design their levels were  $\pm 1$ . In consequence, when solving for  $C$  and  $D$ , we will only consider values within the range  $[-1, 1]$ .

Minimising the Mean Square Error would mean minimising

$$\begin{aligned}MSE(T) &= Var(Y) + (E(Y) - T)^2 \\ &= \exp[-0.59 + 0.33A + 0.34C] \times (12.46 + 0.97A + 1.49D)^2 \\ &\quad + ((12.46 + 0.97 \times A + 1.49)^2 - 144)^2\end{aligned}$$

For various values of  $A$ , the Mean Square Error was minimised and solutions for  $C$  and  $D$  were found subject to the constraint that  $C$  and  $D$  were bounded by  $\pm 1$ . The solutions are plotted against  $A$  in Figure 2.9.

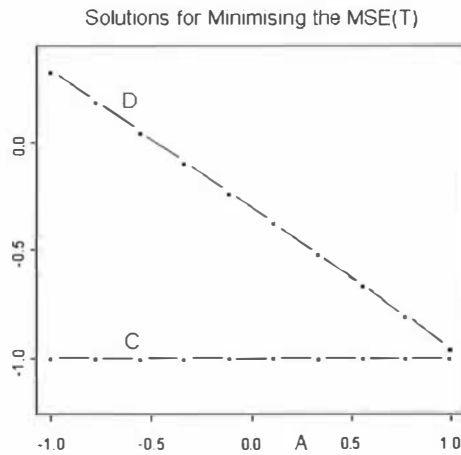


Figure 2.9: The Solution for  $C$  and  $D$  for Different Values of  $A$ .

The solutions observed for  $C$ , for values of  $A$  between  $-1$  and  $1$ , is  $-1$ . The solutions for  $D$  change depending on the values of  $A$ .

If the appropriate software were available then this minimisation problem could be solved every time a new value of  $A$  is observed. Alternatively, regression equations can be used as approximations to the solutions for  $C$  and  $D$ .

The best fit for  $C$  is  $-1$ . Fitting a line, using the observed solutions for  $D$  as the response and  $A$  as the independent variable gives

$$D = -0.3087 - 0.6510A$$

with  $R^2 = 1$ . Therefore with a change in  $A$  the settings for  $D$  could be chosen using that equation.

For this example, the values of  $D$  and  $C$ , given a value of  $A$ , always have the response mean on target. However if the mean was biased for the target then using the adjusted Mean Square Error, as described in the previous section, could be used to move the mean closer to the target.

## 2.7 A Note on Notation

In this chapter three functions have been used in order to find optimal values of the controllable factors. These functions were the Mean Square Error, the adjusted Mean Square Error and the Lagrange function. In the next chapter these

functions will be used again. However there will be no need to differentiate between them and they will all be referred to as variance or adjusted variance functions and labelled  $L(\mathbf{x})$ .

This notation excludes the parameters  $t_i$ . It will be taken that these values have been observed and enter into the function  $L(\mathbf{x})$  as constants.

■

## Chapter 3

# Making Optimal Changes

Observing a change in an observable uncontrollable factor means making a change in the controllable factors in order that some function of the quality variable may be optimised. The best way to change the controllable factors would be to do so instantaneously however there are many practical reasons why this is not possible.

### 3.1 Non-Instantaneous Adjustment

In this case all the controls can be adjusted at once but in a non-instantaneous manner. Suppose that the  $p$  controllable factors have been set at  $\mathbf{x}_0$  after observing the  $q$  observable, uncontrollable factors,  $\mathbf{t}_0$ , at some point prior. At time  $\tau_0 = 0$  a new value of the observable uncontrollable factors,  $\mathbf{t}_1$ , is observed. This requires an adjustment of the controllable factors from  $\mathbf{x}_0$  to  $\mathbf{x}_1$  over some period of time  $\tau_1 - \tau_0 > 0$ .

The way  $\mathbf{x}_1$  is chosen is dependent on the method but usually involves the minimisation of some function,  $L(\mathbf{x})$ , of the mean and variance of  $Y$ , the quality response, as seen in the previous chapter.

There are an infinite number of paths between  $\mathbf{x}_0$  and  $\mathbf{x}_1$  when there is more than one controllable factor. Ideally the best path would be one that minimises the integral of  $L(\mathbf{x})$  taken over the path. In a loose sense this is to minimise the total variance. This lends itself to the methods involved with Calculus of Variations which will be discussed later.

The most obvious path is a straight line between  $\mathbf{x}_0$  and  $\mathbf{x}_1$  and this will be referred to as the straight line solution.

Another way to construct a path is to use a greedy algorithm. In general a greedy algorithm chooses the best option at that particular instant. In this case it would involve heading in the direction that brings about the greatest decrease in  $L(\mathbf{x})$ .

From calculus this direction is given by  $-\frac{\partial L}{\partial \mathbf{x}}$ , evaluated at  $\mathbf{x}$ .

At time,  $\tau = 0$  then  $\mathbf{x} = \mathbf{x}_0$ , the starting point. After some small unit of time we would like  $\mathbf{x}$  to have moved in the direction of steepest descent or, in other words, we want the way  $\mathbf{x}$  changes over time to be equal to  $-\frac{\partial L}{\partial \mathbf{x}}$ . Expressing this mathematically it gives

$$\frac{d\mathbf{x}}{d\tau} = -\frac{\partial L}{\partial \mathbf{x}} \quad (3.1)$$

Unfortunately a greedy algorithm doesn't guarantee an optimal path. However a nice property of the problem is that the greedy algorithm will always construct a path to get to a (local) minimum when that minimum is a critical point. The path that is constructed by this method will be referred to as the greedy path.

### 3.1.1 Explicit Introduction of Unobservable Uncontrollable Factors

As an example consider model 1 the last chapter. It gives

$$L(\mathbf{x}) = \sigma_c^2 + (\boldsymbol{\beta}_3 + \boldsymbol{\Sigma}_2 \mathbf{x} + \boldsymbol{\Sigma}_3 t)^t \boldsymbol{\Lambda}_2 (\boldsymbol{\beta}_3 + \boldsymbol{\Sigma}_2 \mathbf{x} + \boldsymbol{\Sigma}_3 t) + \lambda(T - \beta_0 - \mathbf{x}^t \boldsymbol{\beta}_1 - t^t \boldsymbol{\beta}_2 - \mathbf{x}^t \boldsymbol{\Sigma}_1 t)$$

and that

$$\frac{\partial L}{\partial \mathbf{x}} = 2\boldsymbol{\Sigma}_2^t \boldsymbol{\Lambda}_2 \boldsymbol{\Sigma}_2 \mathbf{x} + 2\boldsymbol{\Sigma}_2^t \boldsymbol{\Lambda}_2 \boldsymbol{\beta}_3 + 2\boldsymbol{\Sigma}_2^t \boldsymbol{\Lambda}_2 \boldsymbol{\Sigma}_3 t - \lambda(\boldsymbol{\beta}_1 + \boldsymbol{\Sigma}_1^t t)$$

or in terms of the previous notation

$$\frac{\partial L}{\partial \mathbf{x}} = 2\mathbf{A}^{-1} \mathbf{x} + 2\mathbf{C} - 2\lambda \mathbf{B}$$

And  $\mathbf{x}^*$  is the solution when  $\frac{\partial L}{\partial \mathbf{x}} = 0$  and a minimum when  $\mathbf{A}$  is positive definite.

To make the notation simpler this will be written as

$$\frac{\partial L}{\partial \mathbf{x}} = -\mathbf{M} \mathbf{x} - \mathbf{N}$$

where

$$\mathbf{M} = -2\mathbf{A}^{-1} \text{ and } \mathbf{N} = -2\mathbf{C} + 2\lambda \mathbf{B}$$

Therefore the path from  $\mathbf{x}_0$  to  $\mathbf{x}_1 = \mathbf{x}^*$  is the solution of

$$\frac{d\mathbf{x}}{d\tau} = \mathbf{M}\mathbf{x} + \mathbf{N}$$

And the solution to this differential equation, a linear system with constant coefficients, is given by

$$\mathbf{x}(\tau) = \exp(\mathbf{M}\tau)\mathbf{x}_0 + \int_0^\tau \exp(\mathbf{M}(\tau - s))\mathbf{N} ds \quad \tau > 0 \quad [10]$$

### Numerical Example

From the numerical example given in the last chapter the underlying model was

$$\begin{aligned} y|\mathbf{x}, \mathbf{t}, \mathbf{z} &= 100 + 5x_1 + 7x_2 - 4t_1 + 3t_2 - 5z_1 - 6z_2 - 4x_1t_1 - 3x_2t_2 + 6z_1x_1 \\ &\quad - 7x_2z_2 - 7x_1z_2 + 8t_1z_1 - 8t_2z_2 + \epsilon \\ &= 100 + \begin{pmatrix} x_1 & x_2 \end{pmatrix} \begin{pmatrix} 5 \\ 7 \end{pmatrix} + \begin{pmatrix} t_1 & t_2 \end{pmatrix} \begin{pmatrix} -4 \\ 3 \end{pmatrix} + \begin{pmatrix} z_1 & z_2 \end{pmatrix} \begin{pmatrix} -5 \\ -6 \end{pmatrix} \\ &\quad + \begin{pmatrix} x_1 & x_2 \end{pmatrix} \begin{pmatrix} -4 & 0 \\ 0 & -3 \end{pmatrix} \begin{pmatrix} t_1 \\ t_2 \end{pmatrix} + \begin{pmatrix} z_1 & z_2 \end{pmatrix} \begin{pmatrix} 6 & 0 \\ -7 & -7 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \\ &\quad + \begin{pmatrix} z_1 & z_2 \end{pmatrix} \begin{pmatrix} 8 & 0 \\ 0 & -8 \end{pmatrix} \begin{pmatrix} t_1 \\ t_2 \end{pmatrix} + \epsilon_i \end{aligned}$$

Suppose that a new observation of  $\mathbf{t}^t = (1, -1)$  is observed and that a path from  $\mathbf{x}_0^t = (1, 1)$  to  $\mathbf{x}^*$  is required. Using Maple the differential equation to be solved is

$$\begin{aligned} \frac{d\mathbf{x}}{d\tau} &= \mathbf{M}\mathbf{x} + \mathbf{N} \\ &= \begin{pmatrix} -170 & -98 \\ -98 & -98 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} -8.1665 \\ 26.3353 \end{pmatrix} \end{aligned}$$

which has as its solution

$$\begin{aligned} \mathbf{x}(\tau) &= \exp(\mathbf{M}\tau)\mathbf{x}_0 + \int_0^\tau \exp(\mathbf{M}(\tau - s))\mathbf{N} ds \\ &= \exp \left[ \begin{pmatrix} -170 & -98 \\ -98 & -98 \end{pmatrix} \tau \right] \begin{pmatrix} 1 \\ 1 \end{pmatrix} \\ &\quad + \int_0^\tau \exp \left[ \begin{pmatrix} -170 & -98 \\ -98 & -98 \end{pmatrix} (\tau - s) \right] \begin{pmatrix} -8.1665 \\ 26.3353 \end{pmatrix} ds \end{aligned}$$

The solution was found using Maple and the output given is where

$\mathbf{x}_0^t = (x_{(0,1)}, x_{(0,2)})$  and the path is  $\mathbf{x}^t = (px, py)$ .

$$\begin{aligned} px &= x_{(0,1)} \frac{1}{2} \exp(-134\tau - 10\sqrt{109}\tau) - x_{(0,1)} \frac{9\sqrt{109}}{545} \exp(-134\tau + 10\sqrt{109}\tau) \\ &\quad + x_{(0,1)} \frac{9\sqrt{109}}{545} \exp(-134\tau - 10\sqrt{109}\tau) + x_{(0,1)} \frac{1}{2} \exp(-134\tau + 10\sqrt{109}\tau) \\ &\quad - x_{(0,2)} \frac{49\sqrt{109}}{1090} \exp(-134\tau + 10\sqrt{109}\tau) + x_{(0,2)} \frac{49\sqrt{109}}{1090} \exp(-134\tau - 10\sqrt{109}\tau) \\ &\quad + .5080034728 \exp(-29.5969349\tau) - .02881203407 \exp(-238.4030651\tau) \\ &\quad - .4791914387 \end{aligned}$$

and

$$\begin{aligned} py &= -x_{(0,1)} \frac{49\sqrt{109}}{1090} \exp(-134\tau + 10\sqrt{109}\tau) + x_{(0,1)} \frac{49\sqrt{109}}{1090} \exp(-134\tau - 10\sqrt{109}\tau) \\ &\quad + x_{(0,2)} \frac{1}{2} \exp(-134\tau - 10\sqrt{109}\tau) + x_{(0,2)} \frac{9\sqrt{109}}{545} \exp(-134\tau + 10\sqrt{109}\tau) \\ &\quad - x_{(0,2)} \frac{9\sqrt{109}}{545} \exp(-134\tau - 10\sqrt{109}\tau) + x_{(0,2)} \frac{1}{2} \exp(-134\tau + 10\sqrt{109}\tau) \\ &\quad - .02011052490 \exp(-238.4030651\tau) - .7278086190 \exp(-29.5969349\tau) \\ &\quad + .7479191439 \end{aligned}$$

for  $\tau > 0$

When  $x_0^t = (1, 1)$  the particular solution is

$$\begin{aligned} px &= 0.3662598616 \exp(-29.5969349 \tau) + 1.649747084 \exp(-238.4030651 \tau) \\ &\quad - 0.4791914387 \end{aligned}$$

$$\begin{aligned} py &= 0.7768156235 \exp(-29.5969349 \tau) - 0.5247347674 \exp(-238.4030651 \tau) \\ &\quad + 0.727808619 \end{aligned}$$

for  $\tau > 0$ .

It is worth noting that the constants at the end of each equation match

$\mathbf{x}^{*t} = (-0.4791914387, 0.727808619)$ . That is, as  $\tau$  tends towards infinity, the curve tends towards  $\mathbf{x}^*$ .

The straight line solution is

$$\mathbf{x} = \mathbf{x}_0 + (\mathbf{x}^* - \mathbf{x}_0)\tau \quad \tau = 0 \dots 1$$

For this example

$$\begin{aligned} \mathbf{x}_1^t &= (1, 1)^t + ((-0.4791914387, 0.727808619)^t - (1, 1))\tau \\ &= (1, 1)^t - (1.48, 0.27)^t \tau \quad \tau = 0 \dots 1 \quad (2dpl) \end{aligned}$$

Figure 3.1 compares the directions of two paths over the surface of  $L$ .

### Alternative Paths after Observing A Change From $\mathbf{x}_0$ to $\mathbf{x}^*$

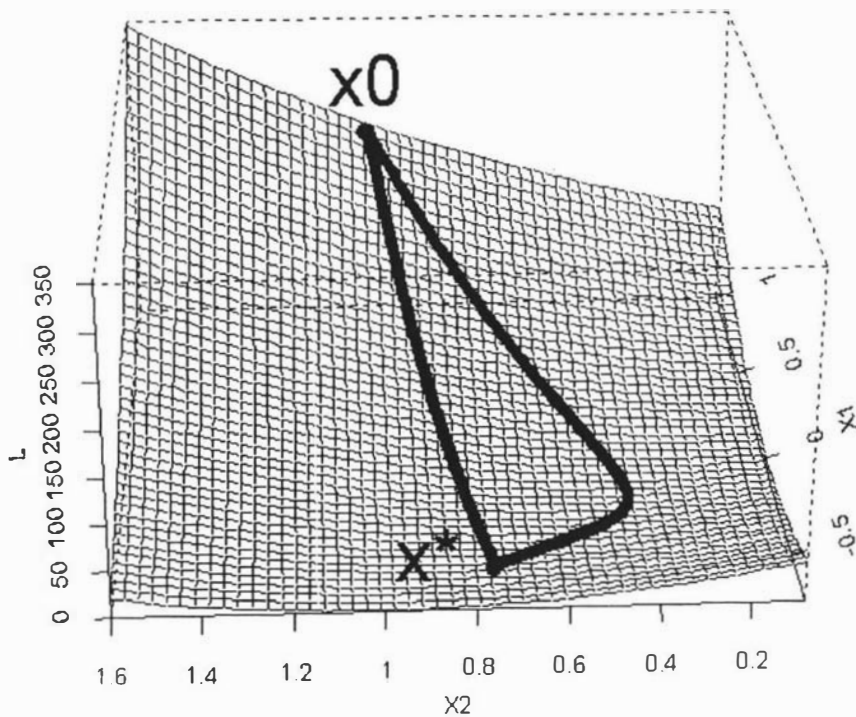
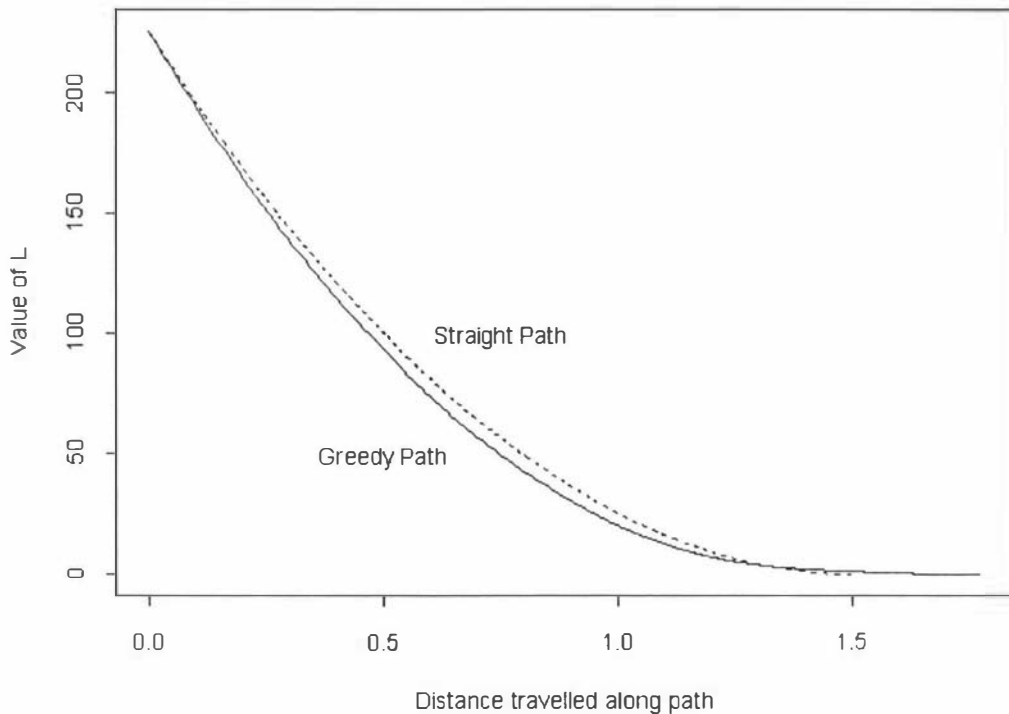


Figure 3.1: A comparison of the directions of the greedy and straight paths

In Figure 3.2 each path has its value of  $L$  plotted against the distance travelled along the path. The direct path to  $\mathbf{x}^*$ , covers the least distance however the greedy path has a smaller value of  $L$  for most of its travelling distance.



The Value of  $L$  versus Distance Travelled for each PathFigure 3.2: A comparison of the value of  $L$  for the two paths

In this example the parameters from the underlying model were used. In practise these parameters would be estimated and used in the equations.

The problem of finding an optimal solution will be further broken down into two cases. When there are two controllable variables and when there are more than two.

### 3.1.2 The Two Variable Case

Suppose  $L(\mathbf{x})$  can be constructed through some means and is a polynomial of order two with two controllable factors with a minimum at some point. Let this polynomial be

$$L_1(x_1, x_2) = b_1x_1^2 + b_2x_1 + b_3x_2^2 + b_4x_2 + b_5x_1x_2 + b_7$$

This function is more complex than necessary for further work and will be transformed to a simpler form. By setting

$$u_1 = x_1 \cos(\theta) + x_2 \sin(\theta)$$

and

$$u_2 = -x_1 \sin(\theta) + x_2 \cos(\theta)$$

and choosing  $\theta$  to be

$$\theta = \cot(2\theta) = \frac{b_1 - b_3}{b_5}$$

then the transformation has rotated  $L_1(x_1, x_2)$  by  $\theta$  in the  $x_1, x_2$  plane.

In the new co-ordinate system the polynomial no longer has a cross-product term and is

$$L_2(u_1, u_2) = a_1 u_1^2 + a_2 u_1 + a_3 u_2^2 + a_4 u_2 + a_7$$

where

$$a_1 = b_1 (\cos(\theta))^2 + b_5 \cos(\theta) \sin(\theta) + b_3 (\sin(\theta))^2$$

$$a_3 := b_1 (\sin(\theta))^2 - b_5 \sin(\theta) \cos(\theta) + b_3 (\cos(\theta))^2$$

$$a_2 := b_2 \cos(\theta) + b_4 \sin(\theta);$$

$$a_4 := -b_2 \sin(\theta) + b_4 \cos(\theta);$$

$$a_7 := b_7;$$

Furthermore by completing the square for  $u_1$  and  $u_2$  the new version of  $L_2$  is

$$L_2(u_1, u_2) = a_1 (u_1 + (a_2/(2a_1)))^2 + a_3 (u_2 + (a_4/(2a_3)))^2 - a_1 (a_2/(2a_1))^2 - a_3 (a_4/(2a_3))^2 + a_7$$

which can be succinctly written as

$$L_3(v_1, v_2) = a_1 v_1^2 + a_3 v_2^2 + a_6$$

In further work it will be sufficient to work with this reduced form.

### 3.1.3 Line Integrals

So far two possible paths have been contemplated, a straight path or a greedy path. The better path to take will be the one that has smaller area under  $L(\mathbf{x})$  along that path. This can be calculated by constructing a line integral for  $L(\mathbf{x})$  along each path.

The line integral for  $L(\mathbf{x})$  is defined to be  $\int_C L(\mathbf{x}) ds$  where  $C$  is the path and  $s$  is an infinitely small vector on  $C$ .

### 3.1.4 Line Integrals For the Greedy and Linear Paths

$L$  in the reduced form is  $L(x, y) = a x^2 + b y^2 + c$ . The requirement is to travel on a path that lies between  $(x_0, y_0)$  and  $(0,0)$ , the minimum of  $L(x, y)$ . It will be assumed that  $L(x, y) \geq 0$ . If it is not then the smallest constant necessary to make it so will be added.

The obvious path is the straight line between these two points which can be algebraically represented by  $y = y_0 t$  and  $x = x_0 t$  for  $t = 1 \dots 0$ . Alternatively this path can be represented as

$$x = x_0 \exp(-et) \text{ and } y = y_0 \exp(-et) \quad t = 0 \dots \infty \quad e > 0$$

An alternative path is the greedy path which can be represented by

$$y = y_0 \exp(-bt) \text{ and } x = x_0 \exp(-at) \quad t = 0 \dots \infty, \quad b, a > 0$$

This was found using the differential equation in the previous section. Without loss of generality let  $b > a$ . If it is not then it just requires an interchange of the labels between  $y$  and  $x$ .

This integral represents the line integral for  $L(x, y) = a x^2 + b y^2 + c$ .

$$\int_{t_0}^{t_1} (ax^2 + by^2 + c)(\dot{x}^2 + \dot{y}^2)^{1/2} |dt|$$

In the following section the line integral for the straight path will be found.

#### Case 1: The Linear Case

The functions of  $x$  and  $y$ , for the linear case, can be represented as

$$x = x_0 \exp(-et) \text{ and } y = y_0 \exp(-et) \quad t = 0 \dots \infty \quad e > 0$$

so that the line integral for this system is

$$\begin{aligned} PI_{linear} &= \int_{t_0}^{t_1} (ax^2 + by^2 + c)(\dot{x}^2 + \dot{y}^2)^{1/2} |dt| \\ &= \int_0^\infty ax^2(\dot{x}^2 + \dot{y}^2)^{1/2} |dt| + \int_0^\infty by^2(\dot{x}^2 + \dot{y}^2)^{1/2} |dt| \\ &\quad + c \int_0^\infty (\dot{x}^2 + \dot{y}^2)^{1/2} |dt| \\ &= \int_0^\infty ax_0^2 \exp(-2et)(e^2 x_0^2 \exp(-2et) + e^2 y_0^2 \exp(-2et))^{\frac{1}{2}} |dt| \\ &\quad + \int_0^\infty by_0^2 \exp(-2et)(e^2 x_0^2 \exp(-2et) + e^2 y_0^2 \exp(-2et))^{\frac{1}{2}} |dt| \end{aligned}$$

$$\begin{aligned}
& +c \int_0^\infty (e^2 x_0^2 \exp(-2et) + e^2 y_0^2 \exp(-2et))^{\frac{1}{2}} |dt| \\
= & ax_0^2 \int_0^\infty (c^2 x_0^2 \exp(-6et) + e^2 y_0^2 \exp(-6et))^{\frac{1}{2}} |dt| \\
& +by_0^2 \int_0^\infty (e^2 x_0^2 \exp(-6et) + e^2 y_0^2 \exp(-6et))^{\frac{1}{2}} |dt| \\
& +c \int_0^\infty (e^2 x_0^2 \exp(-2et) + e^2 y_0^2 \exp(-2et))^{\frac{1}{2}} |dt|
\end{aligned}$$

Now make these substitutions

$$u = e^2 x_0^2 \exp(-6et) + e^2 y_0^2 \exp(-6et)$$

so that

$$du = -6e(e^2 x_0^2 \exp(-6et) + e^2 y_0^2 \exp(-6et)) dt$$

or

$$du = -6eu dt \quad \text{or} \quad \frac{-du}{6eu} = dt \quad \text{and so} \quad |dt| = \frac{|du|}{6eu}$$

and

$$v = e^2 x_0^2 \exp(-2et) + e^2 y_0^2 \exp(-2et)$$

so that

$$dv = -2e(e^2 x_0^2 \exp(-2et) + e^2 y_0^2 \exp(-2et)) dt$$

or

$$dv = -2ev dt \quad \text{or} \quad \frac{-dv}{2ev} = dt \quad \text{and so} \quad |dt| = \frac{|dv|}{2ev}$$

The integral becomes

$$\begin{aligned}
& = ax_0^2 \int_{e^2 x_0^2 + e^2 y_0^2}^0 \frac{u^{\frac{1}{2}}}{6eu} |du| + by_0^2 \int_{e^2 x_0^2 + e^2 y_0^2}^0 \frac{u^{\frac{1}{2}}}{6eu} |du| \\
& \quad +c \int_{e^2 x_0^2 + e^2 y_0^2}^0 \frac{v^{\frac{1}{2}}}{2ev} |dv| \\
& = ax_0^2 \int_{e^2 x_0^2 + e^2 y_0^2}^0 \frac{u^{\frac{1}{2}}}{6eu} (-du) + by_0^2 \int_{e^2 x_0^2 + e^2 y_0^2}^0 \frac{u^{\frac{1}{2}}}{6eu} (-du) \\
& \quad +c \int_{e^2 x_0^2 + e^2 y_0^2}^0 \frac{v^{\frac{1}{2}}}{2ev} (-dv) \\
& = \frac{ax_0^2}{6e} [-2u^{\frac{1}{2}}]_{e^2 x_0^2 + e^2 y_0^2}^0 + \frac{by_0^2}{6e} [-2u^{\frac{1}{2}}]_{e^2 x_0^2 + e^2 y_0^2}^0 + \frac{c}{2e} [-2v^{\frac{1}{2}}]_{e^2 x_0^2 + e^2 y_0^2}^0 \\
& = \frac{ax_0^2 + by_0^2 + 3c}{3e} (e^2 x_0^2 + e^2 y_0^2) \\
& = \frac{ax_0^2 + by_0^2 + 3c}{3} (x_0^2 + y_0^2)
\end{aligned}$$

■

Going back one step in the evaluation of the integral allows for the construction of upper and lower bounds for  $PI_{linear}$ . This is not directly useful here but will be more relevant later. At one step back

$$\begin{aligned}
 PI_{linear} &= \frac{ax_0^2 + by_0^2 + 3c}{3e} (e^2x_0^2 + e^2y_0^2) \\
 &\quad \text{Let } e = b \text{ then} \\
 &\quad \frac{ax_0^2 + by_0^2 + 3c}{3b} (b^2x_0^2 + b^2y_0^2) \\
 &\geq \frac{ax_0^2 + by_0^2 + 3c}{3b} (a^2x_0^2 + b^2y_0^2) \\
 &\quad \text{since } a \leq b \\
 &\quad \text{alternatively let } e = a \text{ then} \\
 PI_{linear} &= \frac{ax_0^2 + by_0^2 + 3c}{3a} (a^2x_0^2 + a^2y_0^2) \\
 &\leq \frac{ax_0^2 + by_0^2 + 3c}{3a} (a^2x_0^2 + b^2y_0^2) \\
 &\quad \text{since } a \leq b
 \end{aligned}$$

This gives the inequality

$$\frac{ax_0^2 + by_0^2 + 3c}{3b} (a^2x_0^2 + b^2y_0^2) \leq PI_{linear} \leq \frac{ax_0^2 + by_0^2 + 3c}{3b} (a^2x_0^2 + b^2y_0^2)$$

## Case 2: The Greedy Case

A closed form solution for the line integral for the greedy path could not be found by the author and an upper and lower bound will be constructed.

The greedy path can be represented by  $y = y_0 \exp(-bt)$  and  $x = x_0 \exp(-at)$  for  $t = 0..∞$  with  $0 < a, b$  and by convention  $a < b$  so that

$$x^2 = x_0^2 \exp(-2at) \text{ and } y^2 = y_0^2 \exp(-2bt)$$

and

$$\dot{x} = -ax_0 \exp(-at) \text{ and } \dot{y} = -by_0 \exp(-bt)$$

and

$$\dot{x}^2 = a^2x_0^2 \exp(-2at) \text{ and } \dot{y}^2 = b^2y_0^2 \exp(-2bt)$$

The path integral is

$$\int_{t_0}^{t_1} (ax^2 + by^2 + c)(\dot{x}^2 + \dot{y}^2)^{1/2} |dt|$$

$$\begin{aligned}
&= \int_0^\infty ax_0^2 \exp(-2at) \left( a^2x_0^2 \exp(-2at) + b^2y_0^2 \exp(-2bt) \right)^{\frac{1}{2}} |dt| \\
&\quad + \int_0^\infty by_0^2 \exp(-2bt) \left( a^2x_0^2 \exp(-2at) + b^2y_0^2 \exp(-2bt) \right)^{\frac{1}{2}} |dt| \\
&\quad + c \int_0^\infty \left( a^2x_0^2 \exp(-2at) + b^2y_0^2 \exp(-2bt) \right)^{\frac{1}{2}} |dt| \\
&= \int_0^\infty ax_0^2 \left( a^2x_0^2 \exp(-6at) + b^2y_0^2 \exp(-2bt - 4at) \right)^{\frac{1}{2}} |dt| \\
&\quad + \int_0^\infty by_0^2 \left( a^2x_0^2 \exp(-2at - 4bt) + b^2y_0^2 \exp(-6bt) \right)^{\frac{1}{2}} |dt| \\
&\quad + c \int_0^\infty \left( a^2x_0^2 \exp(-2at) + b^2y_0^2 \exp(-2bt) \right)^{\frac{1}{2}} |dt|
\end{aligned}$$

Now when there are two constants such that  $0 \leq d < e$  it follows that  $\exp(-dt) \geq \exp(-et)$  for  $t \geq 0$ .

First, by replacing the  $a$ 's with  $b$ 's we have a lower bound for  $PI_{greedy}$ .

$$\begin{aligned}
PI_{greedy} &= \int_0^\infty ax_0^2 \left( a^2x_0^2 \exp(-6at) + b^2y_0^2 \exp(-2bt - 4at) \right)^{\frac{1}{2}} |dt| \\
&\quad + \int_0^\infty by_0^2 \left( a^2x_0^2 \exp(-2at - 4bt) + b^2y_0^2 \exp(-6bt) \right)^{\frac{1}{2}} |dt| \\
&\quad + c \int_0^\infty \left( a^2x_0^2 \exp(-2at) + b^2y_0^2 \exp(-2bt) \right)^{\frac{1}{2}} |dt| \\
&\geq ax_0^2 \int_0^\infty \left( a^2x_0^2 \exp(-6bt) + b^2y_0^2 \exp(-6bt) \right)^{\frac{1}{2}} |dt| \\
&\quad + by_0^2 \int_0^\infty \left( a^2x_0^2 \exp(-6bt) + b^2y_0^2 \exp(-6bt) \right)^{\frac{1}{2}} |dt| \\
&\quad + c \int_0^\infty \left( a^2x_0^2 \exp(-2bt) + b^2y_0^2 \exp(-2bt) \right)^{\frac{1}{2}} |dt|
\end{aligned}$$

Now make these substitutions

$$u = a^2x_0^2 \exp(-6bt) + b^2y_0^2 \exp(-6bt)$$

so that

$$du = -6b(a^2x_0^2 \exp(-6bt) + b^2y_0^2 \exp(-6bt)) dt$$

or

$$du = -6bu dt \quad \text{or} \quad \frac{-du}{6bu} = dt \quad \text{or} \quad |dt| = \frac{|du|}{6bu}$$

and

$$v = a^2x_0^2 \exp(-2bt) + b^2y_0^2 \exp(-2bt)$$

so that

$$dv = -2b(a^2x_0^2 \exp(-2bt) + b^2y_0^2 \exp(-2bt)) dt$$

or

$$dv = -2bv \, dt \quad \text{or} \quad \frac{-dv}{2bv} = dt \quad \text{and so} \quad |dt| = \frac{|dv|}{2bv}$$

which gives

$$\begin{aligned} &= ax_0^2 \int_{a^2x_0^2+b^2y_0^2}^0 \left( \frac{u^{\frac{1}{2}}}{6bu} \right) |du| + by_0^2 \int_{a^2x_0^2+b^2y_0^2}^0 \left( \frac{u^{\frac{1}{2}}}{6bu} \right) |du| \\ &\quad + c \int_{a^2x_0^2+b^2y_0^2}^0 \left( \frac{v^{\frac{1}{2}}}{2bv} \right) |dv| \\ &= ax_0^2 \int_{a^2x_0^2+b^2y_0^2}^0 \left( \frac{u^{-\frac{1}{2}}}{6b} \right) (-du) + by_0^2 \int_{a^2x_0^2+b^2y_0^2}^0 \left( \frac{u^{-\frac{1}{2}}}{6b} \right) (-du) \\ &\quad + c \int_{a^2x_0^2+b^2y_0^2}^0 \left( \frac{v^{\frac{1}{2}}}{2bv} \right) (-dv) \\ &= ax_0^2 \left[ \frac{-2u^{\frac{1}{2}}}{6bu} \right]_{a^2x_0^2+b^2y_0^2}^0 + by_0^2 \left[ \frac{-2u^{\frac{1}{2}}}{6bu} \right]_{a^2x_0^2+b^2y_0^2}^0 \\ &\quad + c \left[ \frac{-2u^{\frac{1}{2}}}{2bv} \right]_{a^2x_0^2+b^2y_0^2}^0 (-dv) \\ &= \frac{1}{3b} (ax_0^2 + by_0^2 + 3c)(a^2x_0^2 + b^2y_0^2) \end{aligned}$$

Secondly, by replacing the  $b$ 's with  $a$ 's we have an upper bound for  $PI_{greedy}$

$$\begin{aligned} PI_{greedy} &= \int_0^\infty ax_0^2 \left( a^2x_0^2 \exp(-6at) + b^2y_0^2 \exp(-2bt - 4at) \right)^{\frac{1}{2}} |dt| \\ &\quad + \int_0^\infty by_0^2 \left( a^2x_0^2 \exp(-2at - 4bt) + b^2y_0^2 \exp(-6bt) \right)^{\frac{1}{2}} |dt| \\ &\quad + c \int_0^\infty \left( a^2x_0^2 \exp(-2at) + b^2y_0^2 \exp(-2bt) \right)^{\frac{1}{2}} |dt| \\ &\leq ax_0^2 \int_0^\infty \left( a^2x_0^2 \exp(-6at) + b^2y_0^2 \exp(-6at) \right)^{\frac{1}{2}} |dt| \\ &\quad + by_0^2 \int_0^\infty \left( a^2x_0^2 \exp(-6at) + b^2y_0^2 \exp(-6at) \right)^{\frac{1}{2}} |dt| \\ &\quad + c \int_0^\infty \left( a^2x_0^2 \exp(-2at) + b^2y_0^2 \exp(-2at) \right)^{\frac{1}{2}} |dt| \end{aligned}$$

Now make these substitutions

$$u = a^2x_0^2 \exp(-6at) + b^2y_0^2 \exp(-6at)$$

so that

$$du = -6a(a^2x_0^2 \exp(-6at) + b^2y_0^2 \exp(-6at)) \, dt$$

or

$$du = -6au \, dt \quad \text{or} \quad \frac{-du}{6au} = dt \quad \text{or} \quad |dt| = \frac{|du|}{6au}$$

and

$$v = a^2x_0^2 \exp(-2at) + b^2y_0^2 \exp(-2at)$$

so that

$$dv = -2a(a^2x_0^2 \exp(-2bt) + b^2y_0^2 \exp(-2bt)) \, dt$$

or

$$dv = -2av \, dt \quad \text{or} \quad \frac{-dv}{2av} = dt \quad \text{and so} \quad |dt| = \frac{|dv|}{2av}$$

which gives

$$\begin{aligned} &= ax_0^2 \int_{a^2x_0^2+b^2y_0^2}^0 \left( \frac{u^{\frac{1}{2}}}{6au} \right) |du| + by_0^2 \int_{a^2x_0^2+b^2y_0^2}^0 \left( \frac{u^{\frac{1}{2}}}{6au} \right) |du| \\ &\quad + c \int_{a^2x_0^2+b^2y_0^2}^0 \left( \frac{v^{\frac{1}{2}}}{2av} \right) |dv| \\ &= ax_0^2 \int_{a^2x_0^2+b^2y_0^2}^0 \left( \frac{u^{-\frac{1}{2}}}{6a} \right) (-du) + by_0^2 \int_{a^2x_0^2+b^2y_0^2}^0 \left( \frac{u^{-\frac{1}{2}}}{6a} \right) (-du) \\ &\quad + c \int_{a^2x_0^2+b^2y_0^2}^0 \left( \frac{v^{-\frac{1}{2}}}{2a} \right) (-dv) \\ &= ax_0^2 \left[ \frac{-2u^{\frac{1}{2}}}{6a} \right]_{a^2x_0^2+b^2y_0^2}^0 + by_0^2 \left[ \frac{-2u^{\frac{1}{2}}}{6a} \right]_{a^2x_0^2+b^2y_0^2}^0 + c \left[ \frac{-2v^{\frac{1}{2}}}{2a} \right]_{a^2x_0^2+b^2y_0^2}^0 \\ &= \frac{1}{3a} (ax_0^2 + by_0^2 + 3c) (a^2x_0^2 + b^2y_0^2) \end{aligned}$$

Putting this together gives

$$\frac{ax_0^2 + by_0^2 + 3c}{3b} (a^2x_0^2 + b^2y_0^2) \leq PI_{greedy} \leq \frac{ax_0^2 + by_0^2 + 3c}{3a} (a^2x_0^2 + b^2y_0^2)$$

Unfortunately both  $PI_{greedy}$  and  $PI_{linear}$  conform to the same bounds so it is not possible to tell from this which one has the lower integral and hence which path is the better path to take.

### 3.1.5 Numerical Comparisons

Since the line integral for the greedy system with two variables has been found impossible to integrate by the author, numerical rather than analytic solutions will have to be compared.



Maple was used to integrate the line integrals for these numerical examples. The first example has  $L(x, y) = 2x^2 + 3y^2$  with  $(7, 2)$  as the initial position. The first path in the example is the straight line path.

```

> x0:=7;
                                x0 := 7
> y0:=2;
                                y0 := 2
> a:=2;
                                a := 2
> b:=3;
                                b := 3
> f:=(a*x^2+b*y^2)*(doty^2+dotx^2)^(1/2);
                                2      2      2      2 1/2
                                ff := (2 xx  + 3 yy ) (dotxx  + dotyy )
> y:=y0*t;
                                y := 2 t
> doty:=diff(y,t);
                                doty := 2
> x:=x0*t;
                                x := 7 t
> dotx:=diff(x,t);
                                dotx := 7
> int1:=evalf(int(f,t=0..1));
                                int1 := 266.9373626

```

The path integral for the straight line case evaluates to 266.94 (2 dpl). The second path is the greedy path.

```

> ff:=(a*xx^2+b*yy^2)*(dotxx^2+dotyy^2)^(1/2);
                                2      2      2      2 1/2
                                ff := (2 xx  + 3 yy ) (dotxx  + dotyy )
> yy:=y0*exp(-b*t);
                                yy := 2 exp(- 3 t)
> dotyy:=diff(yy,t);

```

```

                                dotyy := - 6 exp(- 3 t)
> xx:=x0*exp(-a*t);
                                xx := 7 exp(- 2 t)
> dotxx:=diff(xx,t);
                                dotxx := - 14 exp(- 2 t)
> int2:=evalf(int(ff,t=0..infinity));
                                int2 := 266.366500

```

The path integral for the greedy path evaluates to 266.37 (2dpl). The greedy path produces the smaller path integral and is the preferred path.

The second example has  $L(x, y) = 2x^2 + 3y^2 + 25$  with  $(7, 2)$  as the initial position. The first path in the example is the straight line path.

```

> x0:=7;
                                x0 := 7
> y0:=2;
                                y0 := 2
> a:=2;
                                a := 2
> b:=3;
                                b := 3
> f:=(a*x^2+b*y^2+25)*(doty^2+dotx^2)^(1/2);
                                2      2      2      2 1/2
                                f := (2 x  + 3 y  + 25) (doty  + dotx )
> y:=y0*t;
                                y := 2 t
> doty:=diff(y,t);
                                doty := 2
> x:=x0*t;
                                x := 7 t
> dotx:=diff(x,t);
                                dotx := 7
> int1:=evalf(int(f,t=0..1));
                                int1 := 448.9401098

```

The path integral for the straight line case evaluates to 448.94 (2 dpl). Evaluating the line integral for the greedy path gives

```

> ff:=(a*xx^2+b*yy^2+25)*(dotxx^2+dotyy^2)^(1/2);
                2      2      2      2 1/2
                ff := (2 xx  + 3 yy  + 25) (dotxx  + dotyy )
> yy:=y0*exp(-b*t);
                yy := 2 exp(- 3 t)
> dotyy:=diff(yy,t);
                dotyy := - 6 exp(- 3 t)
> xx:=x0*exp(-a*t);
                xx := 7 exp(- 2 t)
> dotxx:=diff(xx,t);
                dotxx := - 14 exp(- 2 t)
> int2:=evalf(int(ff,t=0..infinity));
                int2 := 449.17177

```

The line integral for the greedy path evaluates to 449.17 (2dpl). The greedy path produces the larger line integral and the straight path is preferred. These two examples show that the preferred path may change from case to case.

The difference in these two examples has been the addition of a constant term to  $L$  i.e. in the first example  $L(x, y) = 2x^2 + 3y^2$  while in the second example  $L(x, y) = 2x^2 + 3y^2 + 25$ . By re-arranging the path integral

$$\begin{aligned}
 PI &= \int_{t_0}^{t_1} (ax^2 + by^2 + c)(\dot{x}^2 + \dot{y}^2)^{1/2} |dt| \\
 &= \int_{t_0}^{t_1} (ax^2 + by^2)(\dot{x}^2 + \dot{y}^2)^{1/2} |dt| + c \int_{t_0}^{t_1} (\dot{x}^2 + \dot{y}^2)^{1/2} |dt|
 \end{aligned}$$

It can be seen that the second integral on the second line is the definition of path length. With a value of  $c > 0$ , minimising the path integral involves, in part, minimising the path length.

## 3.2 On Finding A Minimal Path With Calculus of Variations

Our interest is in finding a path  $C$  with fixed endpoints on a function  $L$  that minimises  $\int_C L ds$ . From the theory on Calculus of Variations[37] there is a result which states that

A necessary condition for  $n$  twice-differentiable functions  $x_1(t), x_2(t), \dots, x_n(t)$  to extremize an integral

$$I = \int_{t_1}^{t_2} f(t, x_1, x_2, \dots, x_n, \dot{x}_1, \dot{x}_2, \dots, \dot{x}_n) dt$$

is that the vector function  $[x_1(t), x_2(t), \dots, x_n(t)]$  be a solution of the  $n$  simultaneous Euler-Lagrange equations

$$\frac{\partial f}{\partial x_i} - \frac{d}{dt} \left( \frac{\partial f}{\partial \dot{x}_i} \right) = 0 \quad i = 1, 2, \dots, n$$

■

Consider the two dimensional case where  $L = L(x, y)$ . The path integral which is to be minimised is

$$\int_C L ds = \int_{t_1}^{t_2} L(x, y) \sqrt{\dot{x}^2 + \dot{y}^2} dt$$

Let  $f(t, x, y, \dot{x}, \dot{y}) = L(x, y) \sqrt{\dot{x}^2 + \dot{y}^2}$  and let  $t_1 < t_2$  then

$$\int_C L ds = \int_{t_1}^{t_2} f(t, x, y, \dot{x}, \dot{y}) dt$$

and this integral is in the form of  $I$  in the theorem.

Now

$$\frac{\partial f}{\partial x} = \frac{\partial L(x, y)}{\partial x} \sqrt{\dot{x}^2 + \dot{y}^2}$$

and

$$\frac{\partial f}{\partial \dot{x}} = L(x, y) \dot{x} (\dot{x}^2 + \dot{y}^2)^{-\frac{1}{2}}$$

and

$$\begin{aligned} \frac{d}{dt} \left( \frac{\partial f}{\partial \dot{x}} \right) &= \frac{d(L(x, y))}{dt} \dot{x} (\dot{x}^2 + \dot{y}^2)^{-\frac{1}{2}} + L(x, y) [\ddot{x} (\dot{x}^2 + \dot{y}^2)^{-\frac{1}{2}} \\ &\quad - \dot{x} (\dot{x} \ddot{x} + \dot{y} \ddot{y}) (\dot{x}^2 + \dot{y}^2)^{-\frac{3}{2}}] \\ &= \dot{x} \frac{\partial L(x, y)}{\partial x} (\dot{x}^2 + \dot{y}^2)^{-\frac{1}{2}} + \dot{y} \frac{\partial L(x, y)}{\partial y} \dot{x} (\dot{x}^2 + \dot{y}^2)^{-\frac{1}{2}} \\ &\quad + L(x, y) [\ddot{x} (\dot{x}^2 + \dot{y}^2)^{-\frac{1}{2}} - \dot{x} (\dot{x} \ddot{x} + \dot{y} \ddot{y}) (\dot{x}^2 + \dot{y}^2)^{-\frac{3}{2}}] \end{aligned}$$

So the first Euler-Lagrange equation becomes

$$\begin{aligned} 0 &= \frac{\partial L(x, y)}{\partial x} (\dot{x}^2 + \dot{y}^2)^{\frac{1}{2}} - \dot{x} \frac{\partial L(x, y)}{\partial x} \dot{x} (\dot{x}^2 + \dot{y}^2)^{-\frac{1}{2}} - \dot{y} \frac{\partial L(x, y)}{\partial y} \dot{x} (\dot{x}^2 + \dot{y}^2)^{-\frac{1}{2}} \\ &\quad - L(x, y) \ddot{x} (\dot{x}^2 + \dot{y}^2)^{-\frac{1}{2}} + L(x, y) \dot{x} (\dot{x} \ddot{x} + \dot{y} \ddot{y}) (\dot{x}^2 + \dot{y}^2)^{-\frac{3}{2}} \end{aligned}$$

$$\begin{aligned}
&= \frac{\partial L(x, y)}{\partial x} (\dot{x}^2 + \dot{y}^2)^2 - \dot{x} \frac{\partial L(x, y)}{\partial x} \dot{x} (\dot{x}^2 + \dot{y}^2) - \dot{y} \frac{\partial L(x, y)}{\partial y} \dot{x} (\dot{x}^2 + \dot{y}^2) \\
&\quad - L(x, y) \ddot{x} (\dot{x}^2 + \dot{y}^2) + L(x, y) \dot{x} (\dot{x} \ddot{x} + \dot{y} \ddot{y}) \\
&= \frac{\partial L(x, y)}{\partial x} (\dot{x}^4 + 2\dot{x}^2 \dot{y}^2 + \dot{y}^4 - \dot{x}^4 - \dot{x}^2 \dot{y}^2) - \dot{y} \frac{\partial L(x, y)}{\partial y} \dot{x} (\dot{x}^2 + \dot{y}^2) \\
&\quad - L(x, y) (\ddot{x} \dot{x}^2 + \ddot{y} \dot{y}^2 - \dot{x}^2 \ddot{x} - \dot{x} \dot{y} \ddot{y}) \\
&= \frac{\partial L(x, y)}{\partial x} (\dot{x}^2 \dot{y}^2 + \dot{y}^4) - \dot{y} \frac{\partial L(x, y)}{\partial y} \dot{x} (\dot{x}^2 + \dot{y}^2) \\
&\quad - L(x, y) (\ddot{x} \dot{y}^2 - \dot{x} \dot{y} \ddot{y}) \\
&= \left( \dot{y} \frac{\partial L(x, y)}{\partial x} - \dot{x} \frac{\partial L(x, y)}{\partial y} \right) (\dot{x}^2 + \dot{y}^2) - L(x, y) (\ddot{x} \dot{y} - \dot{x} \ddot{y})
\end{aligned}$$

or

$$0 = \dot{y}$$

By symmetry, the Euler-Lagrange equation for  $y$  is

$$\begin{aligned}
0 &= \left( \dot{x} \frac{\partial L(x, y)}{\partial y} - \dot{y} \frac{\partial L(x, y)}{\partial x} \right) (\dot{x}^2 + \dot{y}^2) - L(x, y) (\ddot{y} \dot{x} - \dot{y} \ddot{x}) \\
0 &= \dot{x}
\end{aligned}$$

The twice differentiable functions  $x(t), y(t)$  that minimizes the integral  $\int_C L ds$  must satisfy these Euler-Lagrange equations.

■

There are three interesting aspects that come out of these equations.

1) The solutions  $\dot{y} = 0$  and  $\dot{x} = 0$  can be dis-regarded as they are not twice differentiable functions.

2) The first equation

$$\left( \dot{y} \frac{\partial L(x, y)}{\partial x} - \dot{x} \frac{\partial L(x, y)}{\partial y} \right) (\dot{x}^2 + \dot{y}^2) - L(x, y) (\ddot{x} \dot{y} - \dot{x} \ddot{y}) = 0$$

is linearly dependent on its counterpart

$$\left( \dot{x} \frac{\partial L(x, y)}{\partial y} - \dot{y} \frac{\partial L(x, y)}{\partial x} \right) (\dot{x}^2 + \dot{y}^2) - L(x, y) (\ddot{y} \dot{x} - \dot{y} \ddot{x})$$

which means that any solution may not be unique.

3) In the first set of brackets there is this equation

$$\dot{x} \frac{\partial L(x, y)}{\partial y} - \dot{y} \frac{\partial L(x, y)}{\partial x}$$

It is worth noting that this will equal zero when  $x(t)$  and  $y(t)$  are the greedy solutions. For the two variable case, the equation 3.1 has to be solved to get the greedy solution. They are

$$\frac{\partial L(x, y)}{\partial y} = -\dot{y}$$

and

$$\frac{\partial L(x, y)}{\partial x} = -\dot{x}$$

Putting these into the equation above gives

$$\begin{aligned} & \dot{x} \frac{\partial L(x, y)}{\partial y} - \dot{y} \frac{\partial L(x, y)}{\partial x} \\ &= -\dot{x}\dot{y} + \dot{y}\dot{x} \\ &= 0 \end{aligned}$$

Also

$$\ddot{y}\dot{x} - \dot{y}\ddot{x}$$

equals 0 when  $x(t)$  and  $y(t)$  are the straight line solutions. That is if  $x = c_1t + c_2$  and  $y = c_3t + c_4$  then the second derivatives with respect to  $t$  are both zero.

For the case  $L(x, y) = ax^2 + by^2 + c$  there is a set of trivial solutions when  $a = b$ . They are that  $x = x_0t$  and  $y = y_0t$  for  $t = 1 \dots 0$ . It is easy to show this with Maple.

**Theorem 3.1** *The functions  $x = x_0t$  and  $y = y_0t$  for  $t = 1 \dots 0$  are solutions to the Euler-Lagrange equations when  $L(x, y) = ax^2 + ay^2 + c$ .*

**Proof** The Euler-Lagrange equations are

$$\left( \dot{x} \frac{\partial L(x, y)}{\partial y} - \dot{y} \frac{\partial L(x, y)}{\partial x} \right) (\dot{x}^2 + \dot{y}^2) - L(x, y) (\ddot{y}\dot{x} - \dot{y}\ddot{x}) = 0$$

Suppose that  $L(x, y) = ax^2 + by^2 + c$ . The mathematical package Maple will be used to do the calculations for each part of the left hand side of the equation.

```
> y:=y0*t;
```

```
y := y0 t
```

```
> x:=x0*t;
```

```
x := x0 t
```

```
> dotx:=diff(x,t);
```

```
dotx := x0
```

```
> dotdotx:=diff(dotx,t);
```

```

dotdotx := 0
> doty:=diff(y,t);
doty := y0
> dotdoty:=diff(doty,t);
dotdoty := 0
> L:=a*x^2+b*y^2+c;
L := a x0 t 2 + b y0 t 2 + c
> dldy:=2*b*y;
dldy := 2 b y0 t
> dldx:=2*a*x;
dldx := 2 a x0 t

```

These values were fed into the Euler-Lagrange equations (c1+c11).

```

> c1:=expand((dotx*dldy-doty*dldx)*(dotx^2+doty^2)):
> c1:=combine(c1,power):
> c11:=expand((-L)*(dotx*dotdoty-doty*dotdotx)):
> c11:=combine(c11,power):
> c13:=c11+c1:
> factor(c13);

```

$$- 2 x_0 y_0 t (x_0^2 + y_0^2) (-b + a)$$

c13 is the factored Euler-Lagrange equation. Providing  $x_0^2 > 0$  and  $y_0^2 > 0$  then this equation will equal zero when  $b = a$ . This proves that  $x = x_0 t$  and  $y = y_0 t$  for  $t = 1 \dots 0$  are solutions for the case when  $L(x, y) = ax^2 + ay^2 + c$ .

■

However apart from this special case there are no obvious solution. The author tried many functions for  $x(t)$  and  $y(t)$  to solve the Euler-Lagrange equations with no success. A majority of these trial functions were mixtures of the greedy and straight line solution.

The Euler-Lagrange equations can be separated into two parts i.e.

$$\left( \dot{x} \frac{\partial L(x, y)}{\partial y} - \dot{y} \frac{\partial L(x, y)}{\partial x} \right) (\dot{x}^2 + \dot{y}^2)$$

and

$$L(x, y) (\ddot{y}\dot{x} - \dot{y}\ddot{x})$$

The rationale for trying greedy and straight line combinations was that the first part equal zero when  $x(t)$  and  $y(t)$  are the greedy solution. While the second part equals zero when  $x(t)$  and  $y(t)$  are the straight line solution. The hope was that by choosing the right mix the Euler-Lagrange equation would equal zero. ■

There is a second result from the theory of Calculus of Variations[37] that states - *If  $f$  does not involve  $t$  explicitly then the first integral of the system is*

$$\sum_{i=1}^n \dot{x}_i \frac{\partial f}{\partial \dot{x}_i} - f = k$$

Recollecting that  $f = L(x, y)\sqrt{\dot{x}^2 + \dot{y}^2}$  and observing that it doesn't involve  $t$  means that this theorem applies. Now

$$\frac{\partial f}{\partial \dot{x}} = L(x, y)\dot{x}(\dot{x}^2 + \dot{y}^2)^{-\frac{1}{2}}$$

and

$$\frac{\partial f}{\partial \dot{y}} = L(x, y)\dot{y}(\dot{x}^2 + \dot{y}^2)^{-\frac{1}{2}}$$

so the first integral is

$$\begin{aligned} k &= \dot{x}L(x, y)\dot{x}(\dot{x}^2 + \dot{y}^2)^{-\frac{1}{2}} + \dot{y}L(x, y)\dot{y}(\dot{x}^2 + \dot{y}^2)^{-\frac{1}{2}} - L(x, y)\sqrt{\dot{x}^2 + \dot{y}^2} \\ &= L(x, y)\dot{x}^2(\dot{x}^2 + \dot{y}^2)^{-\frac{1}{2}} + \dot{y}^2L(x, y)(\dot{x}^2 + \dot{y}^2)^{-\frac{1}{2}} - L(x, y)\sqrt{\dot{x}^2 + \dot{y}^2} \\ &= \frac{L(x, y)(\dot{x}^2 + \dot{y}^2)}{(\dot{x}^2 + \dot{y}^2)^{\frac{1}{2}}} - L(x, y)\sqrt{\dot{x}^2 + \dot{y}^2} \\ &= 0 \end{aligned}$$

The result collapses into  $k = 0$  which is of no help in finding  $y(t)$  and  $x(t)$ . It is included here only for completeness.

### 3.2.1 Placing Restrictions on the Solution

During personal communications with Professor Graeme Wake of Auckland University he suggested taking a more restrictive relationship between  $x$  and  $y$  namely that  $y = y(x)$  i.e. that  $y$  is a function of  $x$ . Setting  $x = t$  gives

$$\dot{x} = 1 \text{ and } \ddot{x} = 0$$



$$\dot{y} = \frac{dy}{dx} \frac{dx}{dt} = \frac{dy}{dx}$$

and

$$\ddot{y} = \frac{d^2y}{dt^2} = \frac{d}{dt} \left( \frac{dy}{dt} \right) = \frac{d}{dt} \left( \frac{dy}{dx} \right) = \frac{dy'}{dt} = \frac{dy'}{dx} \frac{dx}{dt} = \frac{d^2y}{dx^2}$$

Taking either of the two Euler-Lagrange equations, since they are dependent, and substituting the above results into it gives

$$\begin{aligned} & \left( \dot{x} \frac{\partial L}{\partial y} - \dot{y} \frac{\partial L}{\partial x} \right) (\dot{x}^2 + \dot{y}^2) - L(x, y) (\ddot{y}\dot{x} - \dot{y}\ddot{x}) \\ &= \left( \frac{\partial L}{\partial y} - \frac{dy}{dx} \frac{\partial L}{\partial x} \right) \left( 1 + \left( \frac{dy}{dx} \right)^2 \right) - L(x, y) \frac{d^2y}{dx^2} \end{aligned}$$

which when solving for  $\frac{d^2y}{dx^2}$  gives

$$\frac{d^2y}{dx^2} = \frac{\left( \frac{\partial L}{\partial y} - \frac{dy}{dx} \frac{\partial L}{\partial x} \right) \left( 1 + \left( \frac{dy}{dx} \right)^2 \right)}{L(x, y)}$$

The restriction that  $y = y(x)$  means that for every value of  $x$  there can be only one value for  $y$ . It can't be guaranteed that the function of  $L$  given will have an optimal solution in this form.

Any second order polynomial in two variables with a minimum can be represented in the reduced form  $L(x, y) = a_1x^2 + a_3y^2 + a_6$ . It has the properties that it is symmetric about the  $x$  and  $y$  axes and has a minimum at  $(0,0)$ .

Consider any path from  $|x_0|$  and  $|y_0|$ , to the minimum at  $(0,0)$ . Suppose that an optimal path was constructed by some means and it crosses either or both of the  $x$ -axes and  $y$ -axes. Since the path starts in the positive-positive quadrant and the function is symmetric about the axes then, at the point of crossing, the path can be reflected back into the positive-positive quadrant of  $x$  and  $y$  while still remaining an optimal path. This means that the optimal path can be constructed to always remain in its quadrant of origin.

Now, since there is symmetry, only positive values of  $x_0$  and  $y_0$  need be considered when constructing the optimal path. For if an optimal path from any other signed combination of  $x_0$  and  $y_0$  needed to be constructed, then it need only be constructed from  $|x_0|$  and  $|y_0|$  and reflected in the appropriate axes.

The last thing that will be assumed is that with this  $L$ , the optimal path generated can be represented as a function,  $y(x)$ . The justification for this will be based on the two sub-optimal solutions found. The straight line path gets to the

minimum by taking the path of shortest distance. The greedy path gets to the minimum by taking the path that always gives the lowest value of  $L$  at the next instance. This path isn't restricted by path length and can be as long as it likes. The optimal solution balances both these two objectives. It wants to minimise the heights of  $L$  and take the shortest path. This means it is unlikely to be as curvaceous as the greedy path and will tend to the straight path as  $a_6$  gets very large. Therefore we would expect the optimal path to lie between the greedy path and the straight line path.

Now the straight line path is both a non-decreasing and a non-increasing function. When  $a_1 < a_3$  then the greedy path is an increasing function when going from  $(0,0)$  to  $(|x_0|, |y_0|)$ . If the optimal path lies between these two paths then it seems likely that the optimal path is either an increasing or non-decreasing function.

When  $a_3 < a_1$  then the greedy path is a decreasing function when going from  $(0,0)$  to  $(|x_0|, |y_0|)$ . If the optimal path lies between these two paths then it seems likely that the optimal path is also a decreasing or non-increasing function.

In both cases the optimal path seems likely to be a non-decreasing or non-increasing function i.e. a function. This means placing the restriction of only looking for solutions that have  $y$  as a function of  $x$  seems reasonable.

The problem is now to solve this differential equation

$$\frac{d^2y}{dx^2} = \frac{\left(\frac{\partial L}{\partial y} - \frac{dy}{dx} \frac{\partial L}{\partial x}\right) \left(1 + \left(\frac{dy}{dx}\right)^2\right)}{L(x, y)}$$

with the conditions that  $y(x_0) = y_0$  and  $y(x_1) = y_1$ . One last condition will be that  $y_0$  is not equal to  $y_1$ . This type of problem is called a Boundary Value problem.

For the case  $L(x, y) = a_1x^2 + a_3y^2 + a_6$  the Boundary Value problem is to solve

$$\frac{d^2y}{dx^2} = \frac{(2a_3y - 2a_1xy')(1 + (y')^2)}{a_1x^2 + a_3y^2 + a_6}$$

with  $y(0) = 0$  and  $y(x_0) = y_0$  and  $y_0 > 0$ .

### 3.2.2 The Finite Difference Algorithm

The generic Boundary Value problem can be written as

$$y'' = f(x, y, y'), \quad a \leq x \leq b, \quad y(a) = \alpha, \quad y(b) = \beta, \quad \alpha \text{ not equal to } \beta$$

The finite difference algorithm[11] for non-linear functions of  $y''$  solves this problem providing  $f$  satisfies the following conditions

1)  $f$  and the partial derivatives of  $f_y = \frac{\partial f}{\partial y}$  and  $f_{y'} = \frac{\partial f}{\partial y'}$  are all continuous on

$$D = \{(x, y, y') | a \leq x \leq b, -\infty < y < \infty, -\infty < y' < \infty\}$$

2)  $f_y(x, y, y') \geq \delta > 0$  on  $D$  for some  $\delta > 0$

3) constants  $k$  and  $l$  exist, with

$$k = \max_{(x,y,y') \in D} |f_y(x, y, y')|, \quad l = \max_{(x,y,y') \in D} |f_{y'}(x, y, y')|$$

However if some of these conditions are not met then starting with a good approximation and/or reducing the step size can be considered.

The value of  $f(x, y, y')$  is

$$f(x, y, y') = \frac{\left(\frac{\partial L}{\partial y} - \frac{dy}{dx} \frac{\partial L}{\partial x}\right) \left(1 + \left(\frac{dy}{dx}\right)^2\right)}{L(x, y)}$$

which for  $L(x, y) = a_1x^2 + a_3y^2 + a_6$  is

$$f(x, y, y') = \frac{(2a_3y - 2a_1xy')(1 + (y')^2)}{a_1x^2 + a_3y^2 + a_6}$$

Now

$$f_y(x, y, y') = (1 + (y')^2) \left( \frac{2a_3(a_1x^2 + a_3y^2 + a_6) - (2a_3y - 2a_1xy')2a_3y}{(a_1x^2 + a_3y^2 + a_6)^2} \right)$$

and

$$f_{y'}(x, y, y') = \frac{2y'(2a_3y - 2a_1xy') - (1 + (y')^2)2a_1x}{a_1x^2 + a_3y^2 + a_6}$$

The intention is to reach the minimum at  $(0, 0)$ . When  $a_6 = 0$  the point  $(0, 0)$  means that  $f(x, y, y')$ ,  $f_y(x, y, y')$  and  $f_{y'}(x, y, y')$  are not bounded in any way.

The necessary conditions won't hold.

Also

$$f_y(x, y, y') = (1 + (y')^2) \left( \frac{2a_3(a_1x^2 + a_3y^2 + a_6) - (2a_3y - 2a_1xy')2a_3y}{(a_1x^2 + a_3y^2 + a_6)^2} \right)$$

is required to be greater than zero. With  $x = 0$ ,  $a_6 = 0$  then

$$f_y(x, y, y') = (1 + (y')^2) \left( \frac{2a_3^2y^2 - 4a_3^2y^2}{(a_3y^2)^2} \right) = -(1 + (y')^2) \frac{2}{y^2} < 0$$

and the condition doesn't hold.

This means that the algorithm may not always find the optimal path. From observation, the algorithm fails as the guess for  $y'$  becomes very large and this is most apparent when  $a_1$  is much larger than  $a_3$  or vice versa. From observation the algorithm is much more sensitive to the former case than to the latter. The standard starting approximation is the straight line. Finding a better starting approximation may aid in resolving this problem.

Recall that for  $L(x, y) = a_1x^2 + a_3y^2 + a_6$  the optimal path will be able to be at its longest and most curvaceous when  $a_6 = 0$  and will progressively get shorter as  $a_6$  increases. As  $a_6$  gets very large the optimal solution will tend towards the solution of shortest line length i.e. the straight line. Therefore it is reasonable to assume any optimal solution will be bounded by the optimal solution when  $a_6 = 0$  and the straight line. The aim will be to find a good approximation for  $y$  when  $a_6 = 0$ .

### 3.2.3 Finding an Approximaion

The greedy path will be used as a starting function. Recall that for this greedy system

$$x = x_0 \exp(-a_1 t) \text{ and } y = y_0 \exp(-a_3 t)$$

By eliminating  $t$  this system can be written as

$$y = y_0 \left( \frac{x}{x_0} \right)^{\frac{a_3}{a_1}} \quad x = 0 \dots x_0$$

$\frac{x}{x_0}$  will always be positive as  $x$  and  $x_0$  will always have the same sign i.e. the path always stays in the same quadrant. Now any optimal path can be found by looking at the path from  $|x_0|$  and  $|y_0|$  and doing the appropriate reflections in the axes. So take  $x_0$  and  $y_0$  to be positive.

Taking logs of  $y$  gives

$$\log(y) = \log(y_0) + \frac{a_3}{a_1} \log\left(\frac{x}{x_0}\right)$$

An assumption was made that a good approximation will have a similar form i.e.

$$\log(y) = \log(y_0) - g \log(x_0) + g \log(x)$$

and the intention will be to find  $g$ .

The finite difference algorithm was solved for 4 values of  $a_3$ ,  $a_1$ ,  $x_0$  and  $y_0$  in the ranges

- 1)  $1 < a_3 < 10$ ,
- 2)  $1 < a_1 < a_3$ ,
- 3)  $1 < x_0 < 10$  and
- 4)  $1 < y_0 < 10$  respectively.

These ranges were chosen since the finite difference algorithm was observed to work for these values.

The finite difference algorithm returns a set of values for  $x$  and a set of values for  $y$ . The point  $(x_i, y_i)$  from these sets will represent a point on the optimal path. Logs were taken of  $x$  and  $y$ . Fitting a linear model to the logged data gave reasonably good fits with all  $r^2$ 's being at least 0.95. The variable of interest,  $g$ , is the linear coefficient in this regression. This data is given in appendix E.

The  $g$ 's were then used as a new response variable with  $x_0, y_0, a_1$  and  $a_3$  as the exploratory variables and were contained in the respective columns of the matrix object *pathdetail* in Splus. After much trial and error the function for  $g$  was found to be

$$g = 0.6868 \log\left(\frac{a_3}{a_1}\right) - 0.0360 \left(\log\left(\frac{a_3}{a_1}\right)\right)^2$$

i.e.

```
u<-log(pathdetail[,4]/pathdetail[,3])
ls.print(lsfitt(cbind(u,u^2),log(pathdetail[,6])))
```

Residual Standard Error = 0.0119, Multiple R-Square = 0.9993  
N = 240, F-statistic = 166846 on 2 and 237 df, p-value = 0

	coef	std.err	t.stat	p.value
Intercept	-0.0004	0.0011	-0.3643	0.7159
u	0.6867	0.0035	197.9522	0.0000
	-0.0359	0.0016	-22.0478	0.0000

and has  $r^2 = 0.9993$ . Therefore when  $a_1 < a_3$  a good approximating function is

$$y = y_0 \left(\frac{x}{x_0}\right)^{\exp(0.6867 \log(\frac{a_3}{a_1}) - 0.0359(\log(\frac{a_3}{a_1}))^2)} \quad x = 0 \dots x_0$$

In Figure 3.3 the estimates of  $g$  are plotted against  $g$  and a (0, 1) line has been plotted. The estimate works well for low values of  $g$  but falls off as  $g$  increases. However this seems highly reasonable for use as a starting solution.

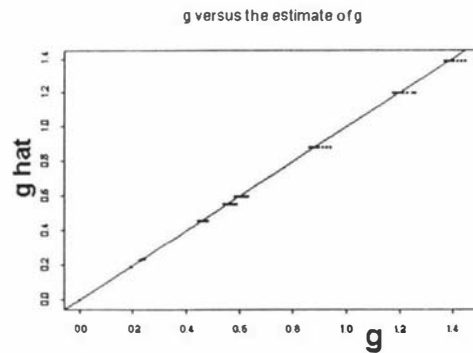


Figure 3.3: The graph of the  $g$ 's versus the estimates of  $g$

With this refinement the finite difference algorithm is complete and is in appendix E. This function is called *optfinddiff.f*. In the same appendix is a function called *transformout.f* that takes a polynomial of order 2 and manipulates it into the reduced form. When  $a_1 > a_3$  this function also interchanges  $x$  with  $y$ , and  $a$  and  $b$  with  $\alpha$  and  $\beta$ , and then calls the finite difference algorithm, *optfinddiff.f*, to find the solution. This latter refinement is to take care of the cases when  $y'$  gets very large which is a by-product when  $a_1 \gg a_3$ . The algorithm can fail under this extreme condition.

After a great deal of use the algorithm, *transformout.f*, has not failed when attempting to produce a path to the minimum of  $L$  when  $L$  has been put into the reduced form. It has also been observed to converge to a solution more quickly than when starting from the straight line solution.

### 3.2.4 Paths to Non-Minimal Points

The paths that have been considered so far have always ended at the minimum of  $L$ . This requirement was necessary in order to derive the greedy path. However it is not a requirement for the construction of a straight line path.

It is also not a requirement for the construction of the optimal path through the numerical method above. An optimal path can be constructed to go to another point besides the minimum. This is particularly useful when there are constraints on the values that the variables can take. For example the variables may be constrained to only take values that were in the space of the original experimental design while the minimum may lie outside this region.

However in justifying that the optimal path could be expressed as a function  $y(x)$ , the properties of the greedy path were used. This justification cannot be used when an end point is not a minimum. It is easy to find end points, that are not minimums, for which an optimal path is not able to be represented as a function  $y(x)$ . In most cases there is a way of getting around this restriction. Consider that we are interested in only the first quadrant. If the path cannot be expressed as a function  $y(x)$ , then by reflecting the surfaces in the line  $y = x$  and re-labelling  $y$  and  $x$  the optimal solution may now have the required form. The only remaining problem is that in the new  $L(x, y) = a_1x^2 + a_3y^2 + a_6$  this property may hold  $a_1 > a_3$ . When  $a_1$  is much greater than  $a_3$  the algorithm may fail. The function *optfinddiff.f*, of the previous section used an approximating function to get a good first approximation for the path to the minimum. This was based on the greedy solution. Since the path of interest doesn't go to the minimum the greedy approximating function is of no use. A straight line as a first approximation will be used. The function, *foptfinddiff.f*, that generates a solution to this boundary value problem is in appendix G. There is also a function called *transformoutnomin.f* that takes a general polynomial of order 2 and puts it in reduced form and then calls *foptfinddiff.f*. It doesn't automatically swap axes as *transformout.f* did. This is because there are now two ways that the algorithm may fail

- 1) if  $a_1 \gg a_3$  or  $a_3 \gg a_1$
- 2) the optimal solution may have  $y' \rightarrow \infty$  on some point on its path

By manipulating the axes manually a solution can be found if one or both of these issues proves to be a problem. Finally the algorithm may fail even with careful manipulation of axes. In that case the best answer is to use a straight line.

### Example

- 1) Consider two different  $L(x, y)$  namely  $L_1 = 5x^2 + y^2$  and  $L_2 = x^2 + 5y^2$  with the starting point being (3,1) and going to (1,1). By using *foptfinddiff.f* the optimal paths can be displayed and are in Figure 3.4.

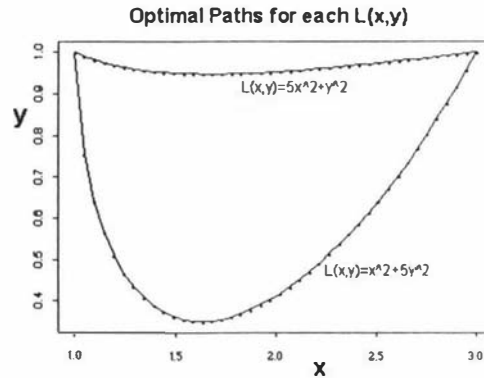


Figure 3.4: The optimal paths for  $L(x, y) = 5x^2 + y^2$  and  $L(x, y) = x^2 + 5y^2$  from  $(3,1)$  to  $(1,1)$ .

2) Consider  $L = 5x^2 + y^2$  with the starting point being  $(3,3)$  and going to  $(4,1)$   
The Splus function call is

```
g1<-foptfinddiff.f(3,4,3,1,40,0.000001,100,5,0,1,0,0,0)
```

For this situation the algorithm fails. This can be seen in Figure 3.5 where the path being generated flips from below the straight line solution to above it. If the algorithm were to continue the paths being generated would continue to flip back and forth without finding a solution. The optimal path has  $y' = \infty$  at some point on the path and the algorithm fails to deal with this.

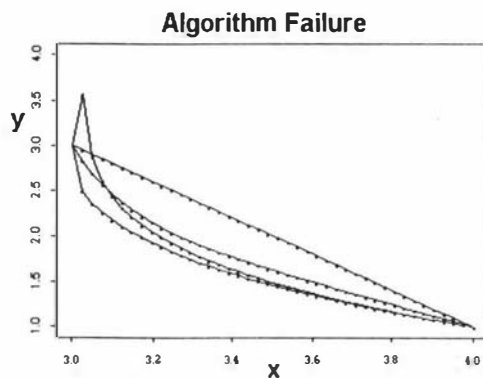


Figure 3.5: The optimal path on  $L = 5x^2 + y^2$  from  $(3,3)$  to  $(4,1)$ .

The solution is to interchange  $x$  with  $y$  and  $a$  and  $b$  with  $\alpha$  and  $\beta$  i.e. the problem becomes  $L = x^2 + 5y^2$  with the starting point being  $(3,3)$  and going to  $(1,4)$ . The Splus function call is



```
g5<-foptfinddiff.f(3,1,3,4,40,0.000001,100,1,0,5,0,0,0)
```

and the paths being generated by the algorithm is in Figure 3.6.

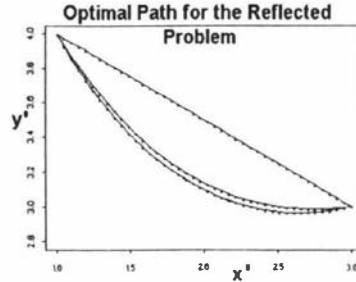


Figure 3.6: The generation of paths that converge to the optimal path on  $L = x^2 + 5y^2$  from  $(3,3)$  to  $(1,4)$ .

Taking the output from the algorithm and re-labelling the output i.e. interchange  $x$  with  $y$  again, gives the optimal solution for the original problem. The path is plotted in Figure 3.7 and it easily observed that  $y$  cannot be represented as a function of  $x$ .

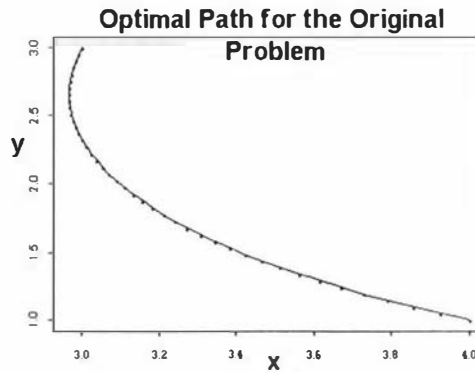


Figure 3.7: The optimal path on  $L = 5x^2 + y^2$  from  $(3,3)$  to  $(4,1)$

### 3.3 Optimal Paths with More than Two Variables

So far the number of controllable factors has been restricted to two. In this section  $p$  controllable factors will be considered. Let  $\mathbf{x}$  represent the  $p$

controllable factors and let  $L(\mathbf{x})$  be the (adjusted) variance of  $\mathbf{x}$  which is a polynomial of order 2. The requirement is to find a path from  $\mathbf{x}_0$  to the minimum of  $L(\mathbf{x})$ ,  $\mathbf{x}^*$ , that minimises the path integral of  $L(\mathbf{x})$ .

The Euler-Lagrange equations can be derived for this system but solutions to these equations were no more obvious to the author than for the 2 dimensional case. The numerical solutions found in the previous section depended on there being only 2 variables in the Euler-Lagrange equations and so that method of solution is not available here.

This leaves three appropriate solutions. The first is to choose a straight line solution i.e.  $\mathbf{x} = \mathbf{x}_0 + (\mathbf{x}^* - \mathbf{x}_0)\tau$ ,  $0 \leq \tau \leq 1$ . This has the benefit of simplicity. The second solution is to compare the line integrals of the straight line path and the greedy path and choose the path with the smaller line integral.

The third solution, if the appropriate software is available, is to consider mixtures of the greedy and straight line solution. The justification for this is based on how the two different solutions come about. The straight line solution is the path of shortest length but doesn't take into consideration the height of  $L(\mathbf{x})$  along its path.

The greedy path doesn't consider path length but aims solely for the maximum decrease in  $L(\mathbf{x})$  at all times.

The optimal solution, which minimises the path integral, looks at minimising the path length and also having low values of  $L(\mathbf{x})$  on the path. This suggests that a mixture of the greedy and straight line solution may make a good approximation to the optimal path.

At point  $\mathbf{x}$  there are two extreme directions that the path can take for a mixture solution. One direction is the path of steepest descent and the other is the direction straight to the minimum. If we take  $\alpha$  units of the first direction and  $1 - \alpha$  units of the second solution we have that

$$\frac{d\mathbf{x}}{d\tau} = -\alpha \frac{\partial L}{\partial \mathbf{x}} + (1 - \alpha)(\mathbf{x}^* - \mathbf{x})$$

With  $L(\mathbf{x})$ , a polynomial of order 2, it gives that  $\frac{dL}{d\mathbf{x}} = \mathbf{M}\mathbf{x} + \mathbf{N}$  so

$$\begin{aligned} \frac{d\mathbf{x}}{d\tau} &= -\alpha \frac{\partial L}{\partial \mathbf{x}} + (1 - \alpha)(\mathbf{x}^* - \mathbf{x}) \\ &= -\alpha(\mathbf{M}\mathbf{x} + \mathbf{N}) + (1 - \alpha)(\mathbf{x}^* - \mathbf{x}) \\ &= (-(1 - \alpha)\mathbf{I} - \alpha\mathbf{M})\mathbf{x} + ((1 - \alpha)\mathbf{x}^* - \alpha\mathbf{N}) \\ &= \mathbf{M}_\alpha \mathbf{x} + \mathbf{N}_\alpha \end{aligned}$$

The solution for this differential equation is

$$\mathbf{x}(\tau) = \exp(\mathbf{M}_\alpha \tau) \mathbf{x}_0 + \int_0^\tau \exp(\mathbf{M}_\alpha(\tau s)) \mathbf{N}_\alpha ds \quad \tau > 0 [10]$$

and this will be the approximating function for the optimal solution. It is worth noting that the approximate function does not satisfy the Euler-Lagrange equations and as such cannot be the optimal solution.

The functions found in this solution are similar in nature to the greedy path i.e. a linear combination of exponentials. An attempt was made by the authour to find closed form solutions for the line integral for a path made of these functions. However no closed form solutions could be found.

### Example

In this example  $L(\mathbf{x})$  will be dependent on two variables so that the results can be compared with previous methods.

Let  $L(x_1, x_2) = x_1^2 + x_2^2 - x_1 x_2$  with the  $\mathbf{x}_0^t = (1, 2)$  and  $\mathbf{x}^{*t} = (0, 0)$ . The Maple program in appendix D was used to generate the mixture path which is

$$x_1 = \frac{3}{2} \exp(-\tau) - \frac{1}{2} \exp(-2\tau a - \tau) \quad \tau \geq 0$$

and

$$x_2 = \frac{1}{2} \exp(-2\tau a - \tau) + \frac{3}{2} \exp(-\tau) \quad \tau \geq 0$$

Now that the path has been found the value for  $\alpha$  will be found that minimises the path integral. This was done in Splus and the code is in appendix D. The value found for  $\alpha$  was  $\alpha = 0.50966$  (5 dpl) which gives the approximate solution, written here to 5 decimal places, as

$$x_1 = \frac{3}{2} \exp(-\tau) - \frac{1}{2} \exp(-2.01935\tau) \quad \tau \geq 0$$

and

$$x_2 = \frac{1}{2} \exp(-2.01935\tau) + \frac{3}{2} \exp(-\tau) \quad \tau \geq 0$$

and the line integral evaluates to 2.15364105 (8 dpl).

However setting  $\alpha = 0$  instead gives the straight line path which is

$$x_1 = \exp(-\tau) \quad \tau \geq 0$$

and

$$x_2 = 2 \exp(-\tau) \quad \tau \geq 0$$

which has a line integral of 2.24 (2 dpl).

Setting  $\alpha = 1$  instead gives the greedy path which is

$$x_1 = \frac{3}{2} \exp(-\tau) - \frac{1}{2} \exp(-3\tau) \quad \tau \geq 0$$

and

$$x_2 = \frac{1}{2} \exp(-3\tau) + \frac{3}{2} \exp(-\tau) \quad \tau \geq 0$$

with a line integral of 2.18 (2 dpl).

The optimal path was generated by the finite difference algorithm and consisted of 1002 points. A Riemann sum approximation to the line integral for this path was calculated with 1001 sub-intervals. This evaluated to 2.15364101 (8 dpl) which is less than the path integral for the greedy case.

The numerical approximation of the line integral for the optimal case is only very slightly less than the numerical approximation of the line integral for the mixture case.

Figure 3.8 compares the three paths: straight, greedy and mixture. The optimal path is not distinguishable from the mixture path at this resolution.

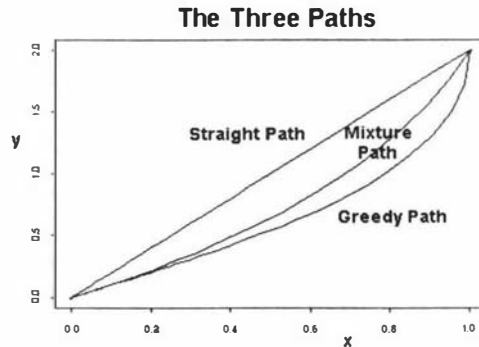


Figure 3.8: A comparison of the three paths: straight, greedy and mixture.

### 3.4 Moving One Factor at a Time

In this situation the goal is to move from an initial setting,  $\mathbf{x}_0$ , of the controllable factors, to a final setting,  $\mathbf{x}_1$ , during some time period  $\tau > 0$ . An added imposition is that the controllable factors can only be moved one factor at a time. In a practical situation this is equivalent to moving the controls by hand when the controls are not within arms length of each other.

This change from  $\mathbf{x}_0$  to  $\mathbf{x}_1$  will be called a transition. Let  $L(\mathbf{x})$  represent a function of the controllable variables and be a variance or adjusted variance function. Each factor change will have an associated path and a corresponding line integral for that path on  $L(\mathbf{x})$ . For  $p$  controllable factors there will be  $p$  paths and so  $p$  line integrals for each transition. The transition that has the smallest value i.e. the smallest sum of line integrals, will be the preferred transition.

### 3.4.1 Two Factor Case

Let  $L(\mathbf{x})$  be a polynomial function of order two, with two controllable factors which can be represented as

$$L(x, y) = ax^2 + bx + cy^2 + dy + exy + f$$

Let  $\mathbf{x}'_0 = (g, h)$  and  $\mathbf{x}'_1 = (i, j)$ . Moving one factor at a time means that there are two paths in each transition from  $\mathbf{x}_0$  to  $\mathbf{x}_1$ . There are also two different transitions available. This is demonstrated in Figure 3.9.

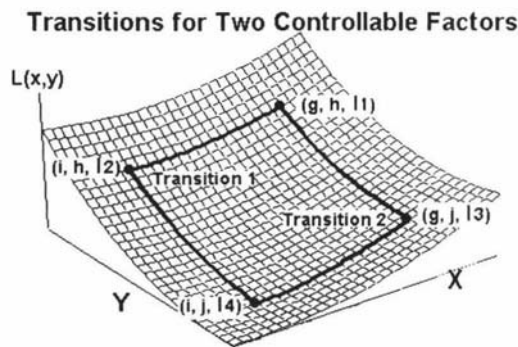


Figure 3.9: A comparison of the two possible transitions and the two paths that make up each transition.

The first transition from  $\mathbf{x}_0$  to  $\mathbf{x}_1$  is to move factor 1,  $x$ , first, from a value of  $g$  to  $i$ , and then factor 2,  $y$ , from a value of  $h$  to  $j$ . This can be represented by  $(g, h) \rightarrow (i, h) \rightarrow (i, j)$ .

The second transition from  $\mathbf{x}_0$  to  $\mathbf{x}_1$  is to move factor 2,  $y$ , first, from a value of  $h$  to  $j$ , and then factor 1,  $x$ , from a value of  $g$  to  $i$ . This can be represented by  $(g, h) \rightarrow (g, j) \rightarrow (i, j)$

### Evaluating the First Transition

The first transition consists of two paths and there is a line integral for each path. The first path is for when factor 1,  $x$ , changes from  $g$  to  $i$  with factor 2,  $y$ , remaining constant at  $h$ . The line integral is

$$T_{(1,1)} = \int_i^g ax^2 + bx + cy^2 + dy + exy + f |dx|$$

If  $i < g$  then

$$\begin{aligned} T_{(1,1)}^+ &= \int_i^g ax^2 + bx + ch^2 + dh + exh + f dx \\ &= \left[ \frac{a}{3}x^3 + \frac{b}{2}x^2 + ch^2x + dhx + \frac{e}{2}hx^2 + fx \right]_i^g \\ &= \frac{a}{3}(g^3 - i^3) + \frac{b}{2}(g^2 - i^2) + ch^2(g - i) + dh(g - i) + \frac{e}{2}h(g^2 - i^2) + f(g - i) \end{aligned}$$

if  $i > g$  then

$$\begin{aligned} T_{(1,1)}^- &= \int_i^g ax^2 + bx + ch^2 + dh + exh + f (-dx) \\ &= \int_g^i ax^2 + bx + ch^2 + dh + exh + f dx \\ &= \left[ \frac{a}{3}x^3 + \frac{b}{2}x^2 + ch^2x + dhx + \frac{e}{2}hx^2 + fx \right]_g^i \\ &= \frac{a}{3}(i^3 - g^3) + \frac{b}{2}(i^2 - g^2) + ch^2(i - g) + dh(i - g) + \frac{e}{2}h(i^2 - g^2) + f(i - g) \end{aligned}$$

The second path is when  $y$  changes from  $h$  to  $j$  with  $x$  remaining constant at its new value  $i$ . The line integral is

$$T_{(1,2)} = \int_j^h ax^2 + bx + cy^2 + dy + exy + f |dy|$$

If  $j < h$  then

$$\begin{aligned} T_{(1,2)}^+ &= \int_j^h ai^2 + bi + cy^2 + dy + eiy + f dy \\ &= \left[ ai^2y + biy + \frac{c}{3}y^3 + \frac{d}{2}y^2 + \frac{e}{2}y^2i + fy \right]_j^h \\ &= ai^2(h - j) + bi(h - j) + \frac{c}{3}(h^3 - j^3) + \frac{d}{2}(h^2 - j^2) + \frac{e}{2}i(h^2 - j^2) + f(h - j) \end{aligned}$$

If  $j > h$  then

$$\begin{aligned} T_{(1,2)}^- &= \int_j^h ai^2 + bi + cy^2 + dy + eiy + f (-dy) \\ &= \left[ ai^2y + biy + \frac{c}{3}y^3 + \frac{d}{2}y^2 + \frac{e}{2}y^2i + fy \right]_h^j \\ &= ai^2(j-h) + bi(j-h) + \frac{c}{3}(j^3-h^3) + \frac{d}{2}(j^2-h^2) + \frac{e}{2}i(j^2-h^2) + f(j-h) \end{aligned}$$

The combined line integrals for this transition is  $T_{(1,1)}^\pm + T_{(1,2)}^\pm$ .

### Evaluating the Second Transition

The second transition also consists of two paths and a line integral for each path. The first path is when  $y$  changes from  $h$  to  $j$  with  $x$  remaining constant at  $g$ . The line integral is

$$T_{(2,1)} = \int_j^h ax^2 + bx + cy^2 + dy + exy + f |dy|$$

If  $j < h$  then

$$\begin{aligned} T_{(2,1)}^+ &= \int_j^h ag^2 + bg + cy^2 + dy + egy + f dy \\ &= \left[ ag^2y + bgy + \frac{c}{3}y^3 + \frac{d}{2}y^2 + \frac{e}{2}y^2g + fy \right]_j^h \\ &= ag^2(h-j) + bg(h-j) + \frac{c}{3}(h^3-j^3) + \frac{d}{2}(h^2-j^2) + \frac{e}{2}g(h^2-j^2) + f(h-j) \end{aligned}$$

If  $j > h$  then

$$\begin{aligned} T_{(2,1)}^- &= \int_j^h ag^2 + bg + cy^2 + dy + egy + f (-dy) \\ &= \int_h^j ag^2 + bg + cy^2 + dy + egy + f dy \\ &= \left[ ag^2y + bgy + \frac{c}{3}y^3 + \frac{d}{2}y^2 + \frac{e}{2}y^2g + fy \right]_h^j \\ &= ag^2(j-h) + bg(j-h) + \frac{c}{3}(j^3-h^3) + \frac{d}{2}(j^2-h^2) + \frac{e}{2}g(j^2-h^2) + f(j-h) \end{aligned}$$

The second path is for when  $x$  changes from  $g$  to  $i$  with  $y$  remaining constant at its new value of  $j$ . The line integral is

$$T_{(2,2)} = \int_i^g ax^2 + bx + cy^2 + dy + exy + f |dx|$$

If  $g > i$  then

$$\begin{aligned} T_{(2,2)}^+ &= \int_i^g ax^2 + bx + cj^2 + dj + exj + f dx \\ &= \left[ \frac{a}{3}x^3 + \frac{b}{2}x^2 + cj^2x + djx + \frac{e}{2}jx^2 + fx \right]_i^g \\ &= \frac{a}{3}(g^3 - i^3) + \frac{b}{2}(g^2 - i^2) + cj^2(g - i) + dj(g - i) + \frac{e}{2}j(g^2 - i^2) + f(g - i) \end{aligned}$$

If  $g < i$  then

$$\begin{aligned} T_{(2,2)}^- &= \int_i^g ax^2 + bx + cj^2 + dj + exj + f (-dx) \\ &= \int_g^i ax^2 + bx + cj^2 + dj + exj + f dx \\ &= \left[ \frac{a}{3}x^3 + \frac{b}{2}x^2 + cj^2x + djx + \frac{e}{2}jx^2 + fx \right]_g^i \\ &= \frac{a}{3}(i^3 - g^3) + \frac{b}{2}(i^2 - g^2) + cj^2(i - g) + dj(i - g) + \frac{e}{2}j(i^2 - g^2) + f(i - g) \end{aligned}$$

The combined line integrals for this transition is  $T_{(2,1)}^\pm + T_{(2,2)}^\pm$

### Comparing Paths

There are four cases corresponding to the values of  $g$ ,  $h$ ,  $i$  and  $j$ .

Case 1:  $g > i$  and  $h > j$ .

The first transition will be preferred if  $T_{(1,1)}^+ + T_{(1,2)}^+ < T_{(2,1)}^+ + T_{(2,2)}^+$  or

$$\begin{aligned} 0 &> T_{(1,1)}^+ + T_{(1,2)}^+ - T_{(2,1)}^+ - T_{(2,2)}^+ \\ &= \frac{a}{3}(g^3 - i^3) + \frac{b}{2}(g^2 - i^2) + ch^2(g - i) + dh(g - i) + \frac{e}{2}h(g^2 - i^2) + f(g - i) \\ &\quad + ai^2(h - j) + bi(h - j) + \frac{c}{3}(h^3 - j^3) + \frac{d}{2}(h^2 - j^2) + \frac{e}{2}i(h^2 - j^2) + f(h - j) \\ &\quad - (ag^2(h - j) + bg(h - j) + \frac{c}{3}(h^3 - j^3) + \frac{d}{2}(h^2 - j^2) + \frac{e}{2}g(h^2 - j^2) + f(h - j)) \\ &\quad - \left( \frac{a}{3}(g^3 - i^3) + \frac{b}{2}(g^2 - i^2) + cj^2(g - i) + dj(g - i) + \frac{e}{2}j(g^2 - i^2) + f(g - i) \right) \\ &= -1/2(-g + i)(-h + j)(ej - 2jc + 2ia - ei + 2b - 2d + eh - 2ch + 2ga - eg) \end{aligned}$$



so the first transition will be preferred when

$$0 < (ej - 2jc + 2ia - ei + 2b - 2d + eh - 2ch + 2ga - eg)$$

or

$$(2a - e)(i + g) + 2b > (2c - e)(h + j) + 2d$$

else the second transition is preferred.

Case 2:  $g > i$  and  $j > h$ .

The first transition will be preferred if  $T_{(1,1)}^+ + T_{(1,2)}^- < T_{(2,1)}^- + T_{(2,2)}^+$  or

$$\begin{aligned} 0 &> T_{(1,1)}^+ + T_{(1,2)}^- - T_{(2,1)}^- - T_{(2,2)}^+ \\ &= 1/2(-g + i)(-h + j)(ej + 2jc + 2ia + ei + 2b + 2d + eh + 2ch + 2ga + eg) \end{aligned}$$

so the first transition will be preferred when

$$0 < ej + 2jc + 2ia + ei + 2b + 2d + eh + 2ch + 2ga + eg$$

or

$$-(2a + e)(i + g) - 2b < (2c + e)(h + j) + 2d$$

else the second transition is preferred.

Case 3:  $g < i$  and  $h > j$ .

The first transition will be preferred if  $T_{(1,1)}^- + T_{(1,2)}^+ < T_{(2,1)}^+ + T_{(2,2)}^-$  or

$$\begin{aligned} 0 &> T_{(1,1)}^- + T_{(1,2)}^+ - T_{(2,1)}^+ - T_{(2,2)}^- \\ &= -1/2(-g + i)(-h + j)(ej + 2jc + 2ia + ei + 2b + 2d + eh + 2ch + 2ga + eg) \end{aligned}$$

so the first transition will be preferred when

$$0 > ej + 2jc + 2ia + ei + 2b + 2d + eh + 2ch + 2ga + eg$$

or

$$-(2a + e)(i + g) - 2b > (2c + e)(h + j) + 2d$$

else the second transition is preferred.

Case 4:  $g < i$  and  $h < j$ .

The first transition will be preferred if  $T_{(1,1)}^- + T_{(1,2)}^- < T_{(2,1)}^- + T_{(2,2)}^-$  or

$$\begin{aligned} 0 &> T_{(1,1)}^- + T_{(1,2)}^- - T_{(2,1)}^- - T_{(2,2)}^- \\ &= 1/2(-g+i)(-h+j)(ej-2jc+2ia-ei+2b-2d+eh-2ch+2ga-eg) \end{aligned}$$

so the first transition will be preferred when

$$0 > ej - 2jc + 2ia - ei + 2b - 2d + eh - 2ch + 2ga - eg$$

or

$$(2a - e)(g + i) + 2b < (2c - e)(h + j) + 2d$$

else the second transition is preferred.

### Example

Let  $L(x, y) = 10 + 3x + 4y + 5x^2 + 4xy + 3y^2$  and let the transition be between  $\mathbf{x}_0^t = (3, 6)$  and  $\mathbf{x}_1^t = (0, 0)$ . This means

$$a = 5, b = 3, c = 3, d = 4, e = 4, f = 10, g = 3, h = 6, i = 0, j = 0$$

Now  $g > i$  and  $h > j$  so it is Case 1.

$$(2a - e)(g + i) + 2b = (2 \times 5 - 4)(3 + 0) + 2 \times 3 = 24$$

and

$$(2c - e)(h + j) + 2d = (2 \times 3 - 4)(6 + 0) + 2 \times 4 = 20$$

Since  $24 > 20$  then  $(2a - e)(g + i) + 2b > (2c - e)(h + j) + 2d$  and the first transition is preferred i.e. moving factor 1,  $x$ , from 3 to 0 and then factor 2,  $y$ , from 6 to 0.

If  $\mathbf{x}_0^t = (3, 8)$  then  $(2a - e)(g + i) + 2b = (2c - e)(h + j) + 2d = 24$  and either transition is preferred.

If  $\mathbf{x}_0^t = (3, 9)$  then  $24 = (2a - e)(g + i) + 2b < (2c - e)(h + j) + 2d = 26$  and the second transition is preferred.

### 3.4.2 Two or More Factors Case

If the number of controllable factors is  $p \geq 2$  then there are  $p!$  possible transitions. For each transition there are  $p$  line integrals to be evaluated. As  $p$  gets large this is a problem most suited to be solved by computer rather than by hand.

Let  $L(\mathbf{x})$  be a polynomial function of order 2 with  $p$  controllable factors which can be represented as

$$L(\mathbf{x}) = A + \mathbf{x}^t \mathbf{B} + \mathbf{x}^t \mathbf{C} \mathbf{x}$$

Where  $C$  is a symmetric matrix. Let  $\mathbf{x}_0$  be the initial position and  $\mathbf{x}_1$  be the final position. The algorithm *whichtochange.f*, in appendix I, takes this data and returns the transition with the smallest value. If there is more than 1 transition that has the smallest value then all the transitions with that smallest value will be returned.

### Example

Using the data in the previous example gives

$$\begin{aligned} L(\mathbf{x}) &= A + \mathbf{x}^t \mathbf{B} + \mathbf{x}^t \mathbf{C} \mathbf{x} \\ &= 10 + (x, y) \begin{pmatrix} 3 \\ 4 \end{pmatrix} + (x, y) \begin{pmatrix} 5 & 2 \\ 2 & 3 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \end{aligned}$$

and  $\mathbf{x}_0^t = (3, 6)$  and  $\mathbf{x}_1^t = (0, 0)$ . Considering both transitions and using the algorithm *whichtochange.f* gives that the first transition is best with the sum of path integrals for each transition being 940.5 and 976.5 respectively.

When  $\mathbf{x}_0^t = (3, 8)$  then either transition is preferred with the sum of path integrals for both transitions being 1624.5.

When  $\mathbf{x}_0 = (3, 9)$  then the second transition is preferred with the sum of path integrals for each transition being 2068.5 and 2041.5 respectively.

## 3.5 Moving Two Factors at a Time

In this situation the goal is to move from an initial setting,  $\mathbf{x}_0$ , of the controllable factors, to a final setting,  $\mathbf{x}_1$ , during some time period  $\tau > 0$ . An added imposition will be that the controllable factors can only be moved two factors at a time. In a practical situation moving two factors at a time is equivalent to moving the controls when the operator can use both hands.

With an even number of factors then all the moves can be two factor moves. With an odd number of factors there will be a single factor move. This single factor move can occur at any stage, before, after or between any of the two factor moves.

This change from  $\mathbf{x}_0$  to  $\mathbf{x}_1$  will be called a transition and consist of the two factor and possible one factor moves. Let  $L(\mathbf{x})$  represent a function of the controllable variables and be a variance or adjusted variance function. Each two factor or single factor change will have an associated path and a corresponding line integral for that path on  $L(\mathbf{x})$ . The transition that has the smallest value i.e. the smallest sum of line integrals for the factor changes, will be the preferred transition.

With  $p$  controllable factors there will be  $n_p$  unique transitions. If  $p$  is even then there are  $n_p = \frac{p!}{2^{\frac{p}{2}}}$  transitions. For the case when  $p$  is odd the number of transitions is more complex to work out. If the single factor move is always last then there will be  $\frac{p!}{2^{\frac{p-1}{2}}}$  transitions. However the single factor transition can be first, last or between any pair of factors and there are  $\frac{(p+1)}{2}$  ways of arranging that amongst the pairs. This gives  $n_p = \frac{(p+1)!}{2^{\frac{p+1}{2}}}$  transitions in total.

There are an infinite number of paths that can be taken between two points in two dimensions. However straight line paths will be the only paths considered for these two factor changes. The cost in time of getting accurate values for the line integrals for the optimal paths, as considered in the last section, would swamp any benefit in using them i.e. while waiting for the best transition to be calculated the controllable factors remain at a non-optimal setting.

Once the minimal transition is found then the choice of paths i.e. straight, greedy or optimal, can be chosen according to which is the most practical.

### 3.5.1 Two or More Factors Case

Let  $L(\mathbf{x})$  be a polynomial function of order 2 with  $p$  controllable factors which can be represented as

$$L(\mathbf{x}) = A + \mathbf{x}^t \mathbf{B} + \mathbf{x}^t \mathbf{C} \mathbf{x}$$

Where  $\mathbf{C}$  is a symmetric matrix. Let  $\mathbf{x}_0$  be the initial position and  $\mathbf{x}_1$  be the final position. There are two cases to consider, when the number of factors is even and when the number of factors is odd. The former case is very much simpler and the algorithm that finds the minimal transition is *whichtwotochangeeven.f* and is in Appendix J.

The latter case, when there is an odd number of factors means that single factor moves need to be considered. This is computationally much harder. The algorithm for this is also in Appendix J and is called *whichtwotochangeodd.f*

In both algorithms, if there is more than one minimal transition, then all those minimal transitions are reported.

### Example

1) For this first case there will be an even number of factors. Consider this value of  $L(\mathbf{x})$

$$L(x_1, x_2, x_3, x_4) = x_1^2 + 2x_2^2 + 3x_3^2 + 4x_4^2$$

with  $\mathbf{x}_0^l = (5, 5, 5, 5)$  and  $\mathbf{x}_1^l = (-1, -1, -1, -1)$ .

This gives

$$A = 0 \quad B = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \quad C = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 4 \end{pmatrix}$$

From observation the minimal transition would be to move factors 3 and 4 first, then factors 1 and 2.

The matrix, *paths*, of possible transitions is listed below. Each row represents a transition. The fifth row represents the transition, move factors 2 and 4 first and then factors 1 and 3.

1	2	3	4
1	3	2	4
1	4	2	3
2	3	1	4
2	4	1	3
3	4	1	2

With  $p = 4$  factors there are  $n_p = \frac{p!}{2^{\frac{p}{2}}} = 6$  transitions. They are all accounted for in *paths*.

Running *wichtwotochangeeven.f* gives this output

```
1 2104.34978081116
2 1900.70302782944
3 1697.05627484771
4 1697.05627484771
5 1493.40952186599
```

6 1289.76276888426

\$paths:

[1] 6

\$min.path.length:

[,1]

[1,] 1289.763

The output says that row 6 of the *paths* matrix gives the minimal transition. This is move factors 3 and 4 first then factors 1 and 2. This transition has value 1289.76.

2) In this case there are an odd number of factors. Consider this value of  $L(\mathbf{x})$

$$L(x_1, x_2, x_3, x_4, x_5) = x_1^2 + 2x_2^2 + 3x_3^2 + 4x_4^2 + 5x_5^2$$

with  $\mathbf{x}_0^t = (5, 5, 5, 5, 5)$  and  $\mathbf{x}_1^t = (-1, -1, -1, -1, -1)$ .

This gives

$$A = 0 \quad B = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \quad C = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 3 & 0 & 0 \\ 0 & 0 & 0 & 4 & 0 \\ 0 & 0 & 0 & 0 & 5 \end{pmatrix}$$

From observation the minimal transition would be to move factors 4 and 5 first, then factors 2 and 3 then factor 1.

The matrix *paths*, in this case, represents all the possible transitions where the last factor is the factor that moves singly. During computation the other possible single factor moves will be considered. For example, the first row of *paths* represents the transition, move factors 1 and 2, then factors 3 and 4 and finally factor 5. During computation the transition that has factor 5 first, and the transition that has factor 5 move occurring between the two factor moves will also be evaluated. Putting *paths* in this 'short-hand' notation is to ease computation and to reduce *paths* size. Here the matrix *paths* has been formatted over two columns.

Column 1	Column 2
1 2 3 4 5	2 4 1 3 5
1 2 3 5 4	2 4 1 5 3
1 2 4 5 3	2 4 3 5 1
1 3 2 4 5	2 5 1 3 4
1 3 2 5 4	2 5 1 4 3
1 3 4 5 2	2 5 3 4 1
1 4 2 3 5	3 4 1 2 5
1 4 2 5 3	3 4 1 5 2
1 4 3 5 2	3 4 2 5 1
1 5 2 3 4	3 5 1 2 4
1 5 2 4 3	3 5 1 4 2
1 5 3 4 2	3 5 2 4 1
2 3 1 4 5	4 5 1 2 3
2 3 1 5 4	4 5 1 3 2
2 3 4 5 1	4 5 2 3 1

As  $p = 5$  the number of transitions with the single factor movement happening last is  $\frac{5!}{2^{\frac{5-1}{2}}} = 30$  and these are all listed in *paths*. To represent all the possible transitions would require a matrix of length  $\frac{6!}{2^{\frac{6}{2}}} = 90$  rather than 30.

The truncated output from *whichtwotochangeodd.f* using the data above is

```

1 Singleton first 3899.20259455355
1 Singleton before pair 2 4485.43635946218
1 Singleton last 4495.67012437081
2 Singleton first 3695.55584157183
2 Singleton before pair 2 4137.78960648045
2 Singleton last 4292.02337138908
3 Singleton firs ....
      ...last 2707.58441227157
30 Singleton first 4127.55584157182
30 Singleton before pair 2 3035.20259455355
30 Singleton last 2518.84934753528
$paths:
      [,1] [,2]
[1,] 30 3
```

\$min.path.length:

[,1]

[1,] 2518.849

The program concludes that the thirtieth path gives that smallest transition value and that the single factor move should come before the third pair. Since there are only two pairs this means the single factor move comes last. This transition is move factors 4 and 5 first, then move factors 2 and 3 and finally move factor 1. This transition has a value of 2518.849.

■



## Chapter 4

# On-Line Quality Control

During production the value of the quality control variable,  $Y$ , is often recorded on quality control charts. The most basic chart, the X-chart, consists of two lines about  $\mu_Y$  at a distance of  $\pm 3\sigma_Y$ . Any point that lies outside these lines is suspect and is either deemed indicative of a process out of control or an observation that has happened purely by chance. Rules of thumb, based on observed sequences of these suspect points, are used to make a decision on the stability of the process. However the quality variable is just a reflection of the error variable. Whenever the former is out of control then it is the error variable that is out of control. When there are more sources of randomness than the error distribution alone i.e. the uncontrollable variables, observing the quality control variable is not going to be a reliable measure of whether any of the random variables are out of control. For example say that  $Y = Z + \epsilon$  where  $Z$  and  $\epsilon$  are independent and bivariate standard Normal. This means that  $Y$  has a  $N(0,2)$  distribution and would have control limits set at  $\pm 3\sigma_Y = \pm 6$ . After some time this set of  $Y$  values may be observed

Run	$Z$	$\epsilon$	$Y$
1	3.6884570	0.8666916	4.5551486
2	-6.6419924	0.3272760	-6.3147164
3	-3.6477334	-0.8147046	-4.4624380
4	3.5191568	0.1508901	3.6700469

Now  $Y$  is observed to be outside of the control limits at at run 2. If this were the only such occurrence for some considerable time then it would most likely be deemed to have happened by chance alone. However it appears that  $Z$  may have

had a change in distribution (specifically a change in variance) and is outside the  $\pm 3\sigma_z$  limits for all of these runs. If this information about  $Z$  had been available then it would have aided in realising that the system was now out of control. After more runs the out of control behaviour would most likely be spotted. However there would be no indication as to which of the variables,  $Z$  or  $\epsilon$  was the one causing the out of control behaviour. When there are many variables contributing to the variance, quickly identifying which variable is causing the out of control behaviour will aid in getting the system back under control more quickly.

The problem is that the unobservable, uncontrollable variables, as their name suggests, are unobservable so the best that can be done is to estimate them.

## 4.1 The Distributions of $Z$ and $\epsilon$ Given $Y$

Suppose that we have the following information. The unobservable uncontrollable variables,  $Z$ , are distributed  $N_r(\mathbf{0}_r, \Lambda_2)$  where  $\mathbf{0}_r$  is a matrix of size  $r \times 1$  of zeros. For simplicity  $\Lambda_2$  will be written as  $\Lambda$  and has elements

$$\Lambda[i, j] = \begin{cases} \sigma_i^2 & \text{for } i = j \\ \rho_{i,j} \sigma_i \sigma_j & i \neq j \end{cases}$$

The pure random error,  $\epsilon$ , is distributed as  $N(0, \sigma_\epsilon^2)$ . Now let  $\epsilon$  and  $Z$  be independent and have a joint distribution

$$\begin{pmatrix} Z \\ \epsilon \end{pmatrix} \sim N_{r+1} \left( \begin{pmatrix} \mathbf{0}_r \\ 0 \end{pmatrix}, \begin{pmatrix} \Lambda & \mathbf{0}_r \\ \mathbf{0}_r^t & \sigma_\epsilon^2 \end{pmatrix} \right)$$

Let the quality variable,  $Y$ , be related to  $Z$  and  $\epsilon$  by the equation

$$Y = A + Z^t B + \epsilon$$

Thus for example, for model 1 in chapter 2, we have that

$$A = \beta_0 + \mathbf{x}^t \beta_1 + t^t \beta_2 + \mathbf{x}^t \Sigma_1 t$$

and

$$B = \beta_3 + \Sigma_2 \mathbf{x} + \Sigma_3 t$$

An estimate of  $Z$  and  $\epsilon$  will be found by considering the distributions of  $Z|Y$  and  $\epsilon|Y$ . Finding the distribution of  $Z_i|Y$  will be done in three steps

- 1) the joint distribution of  $Z$  and  $Y$  will be found,
- 2) the marginal distribution of  $Z_i$  and  $Y$  will be found and finally
- 3) the conditional distribution of  $Z_i|Y = y$  will be found.

**Theorem 4.1** *The joint distribution of  $Z$  and  $Y$  is*

$$\begin{pmatrix} Z \\ Y \end{pmatrix} \sim N_{r+1} \left( \begin{pmatrix} \mathbf{0}_r \\ A \end{pmatrix}, \begin{pmatrix} \Lambda & \Lambda B \\ B^t \Lambda & B^t \Lambda B + \sigma_\epsilon^2 \end{pmatrix} \right)$$

**Proof**

A transformation will be used to get the joint distribution of  $Y - A$  and  $Z$  from the joint distribution of  $Z$  and  $\epsilon$ .

There is a result which says that if  $V$  is  $N_n(\mu, \Sigma)$  and  $W = CV$  where  $C$  is a matrix of constants with dimensions  $n \times n$  and  $\text{rank}=n$  then  $W$  is  $N_n(C\mu, C\Sigma C^t)$ . [38]

Now

$$\begin{pmatrix} Z \\ Y - A \end{pmatrix} = \begin{pmatrix} I_r & \mathbf{0}_r \\ B^t & 1 \end{pmatrix} \begin{pmatrix} Z \\ \epsilon \end{pmatrix}$$

The transformation matrix is  $C = \begin{pmatrix} I_r & \mathbf{0}_r \\ B^t & 1 \end{pmatrix}$  and is  $(r + 1) \times (r + 1)$ .

$C$  contains  $I_r$  which is  $r \times r$  so it has rank of at least  $r$ . Let the  $i^{th}$  element of  $B$  be  $b_i$  of which there are  $r$  such elements. Perform these row operations on the last row i.e. row  $(r + 1)$  that contains  $B^t$

$$\begin{aligned} \text{row } (r + 1) &= \text{row } (r + 1) - \frac{1}{b_1} \times \text{row } 1 \\ &\vdots \\ \text{row } (r + 1) &= \text{row } (r + 1) - \frac{1}{b_i} \times \text{row } i \\ &\vdots \\ \text{row } (r + 1) &= \text{row } (r + 1) - \frac{1}{b_r} \times \text{row } r \end{aligned}$$

Now the row reduced version of  $C$ ,  $C_{\text{reduced}}$ , is an identity matrix of size  $r + 1$  and so has  $\text{rank}(C_{\text{reduced}}) = r + 1$  and thus  $C$  is of full rank.

The mean of  $\begin{pmatrix} Z \\ Y - A \end{pmatrix}$  is

$$C\mu = \begin{pmatrix} I_r & 0_r \\ B^t & 1 \end{pmatrix} \begin{pmatrix} 0_r \\ 0 \end{pmatrix} = \begin{pmatrix} 0_r \\ 0 \end{pmatrix}$$

and the variance is

$$\begin{aligned} C\Sigma C^t &= \begin{pmatrix} I_r & 0_r \\ B^t & 1 \end{pmatrix} \begin{pmatrix} \Lambda & 0_r \\ 0_r^t & \sigma_\epsilon^2 \end{pmatrix} \begin{pmatrix} I_r & 0_r \\ B^t & 1 \end{pmatrix}^t \\ &= \begin{pmatrix} \Lambda & 0_r \\ B^t\Lambda & \sigma_\epsilon^2 \end{pmatrix} \begin{pmatrix} I_r & B \\ 0_r^t & 1 \end{pmatrix} \\ &= \begin{pmatrix} \Lambda & \Lambda B \\ B^t\Lambda & B^t\Lambda B + \sigma_\epsilon^2 \end{pmatrix} \end{aligned}$$

hence

$$\begin{pmatrix} Z \\ Y - A \end{pmatrix} \sim N_{r+1} \left( \begin{pmatrix} 0_r \\ 0 \end{pmatrix}, \begin{pmatrix} \Lambda & \Lambda B \\ B^t\Lambda & B^t\Lambda B + \sigma_\epsilon^2 \end{pmatrix} \right)$$

Now

$$\begin{pmatrix} Z \\ Y \end{pmatrix} = \begin{pmatrix} Z \\ Y - A \end{pmatrix} + \begin{pmatrix} 0_r \\ A \end{pmatrix}$$

so

$$\begin{pmatrix} Z \\ Y \end{pmatrix} \sim N_{r+1} \left( \begin{pmatrix} 0_r \\ A \end{pmatrix}, \begin{pmatrix} \Lambda & \Lambda B \\ B^t\Lambda & B^t\Lambda B + \sigma_\epsilon^2 \end{pmatrix} \right)$$

■

**Theorem 4.2** *The marginal distribution of  $Z_i$  and  $Y$  is*

$$\begin{pmatrix} Z_i \\ Y \end{pmatrix} \sim N_2 \left( \begin{pmatrix} 0 \\ A \end{pmatrix}, \begin{pmatrix} \sigma_i^2 & \sum_{j=1, j \neq i}^n B_j \rho \sigma_j \sigma_i + B_i \sigma_i^2 \\ \sum_{j=1, j \neq i}^n B_j \rho \sigma_j \sigma_i + B_i \sigma_i^2 & B^t \Lambda B + \sigma_\epsilon^2 \end{pmatrix} \right)$$

**Proof**

There is a result that says if  $V$  is  $N_p(\mu, \Sigma)$  and  $C$  is a  $q \times p$  matrix of constants with  $\text{rank}(C) = q (\leq p)$  then if  $W = CV$  then  $W$  is  $N_q(C\mu, C\Sigma C^t)$ .<sup>[38]</sup>

Let  $C_i$  be a  $2 \times (r + 1)$  matrix where

$$C_i[j, k] = \begin{cases} 1 & j = 1, k = i, k \neq r + 1 \\ 1 & j = 2, k = r + 1 \\ 0 & \text{otherwise} \end{cases}$$

The  $C_i$  matrix has rank 2 and picks the  $i^{th}$  and  $(r + 1)^{th}$  rows of  $\begin{pmatrix} Z \\ Y \end{pmatrix}$  to give the marginal distribution of  $Z_i$  and  $Y$  i.e.

$$C_i \begin{pmatrix} Z \\ Y \end{pmatrix} = \begin{pmatrix} 0 & \dots & 0 & 1 & 0 & \dots & 0 & 0 \\ 0 & \dots & 0 & 0 & 0 & \dots & 0 & 1 \end{pmatrix} \begin{pmatrix} Z \\ Y \end{pmatrix} = \begin{pmatrix} Z_i \\ Y \end{pmatrix}$$

The mean of  $Z_i$  and  $Y$  is

$$C_i \mu = \begin{pmatrix} 0 & \dots & 0 & 1 & 0 & \dots & 0 & 0 \\ 0 & \dots & 0 & 0 & 0 & \dots & 0 & 1 \end{pmatrix} \begin{pmatrix} \mathbf{0}_r \\ A \end{pmatrix} = \begin{pmatrix} 0 \\ A \end{pmatrix}$$

and the variance is

$$\begin{aligned} C_i \Sigma C_i^t &= \begin{pmatrix} 0 & \dots & 0 & 1 & 0 & \dots & 0 & 0 \\ 0 & \dots & 0 & 0 & 0 & \dots & 0 & 1 \end{pmatrix} \begin{pmatrix} \Lambda & \Lambda B \\ B^t \Lambda & B^t \Lambda B + \sigma_\epsilon^2 \end{pmatrix} \begin{pmatrix} 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \\ 0 & 1 \end{pmatrix} \\ &= \begin{pmatrix} \rho \sigma_i \sigma_1 \dots \sigma_i^2 \dots \rho \sigma_i \sigma_r & \sum_{j=1, j \neq i}^r B_j \rho \sigma_j \sigma_i + B_i \sigma_i^2 \\ B^t \Lambda & B^t \Lambda B + \sigma_\epsilon^2 \end{pmatrix} \begin{pmatrix} 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \\ 0 & 1 \end{pmatrix} \\ &= \begin{pmatrix} \sigma_i^2 & \sum_{j=1, j \neq i}^r \rho B_j \sigma_j \sigma_i + B_i \sigma_i^2 \\ \sum_{j=1, j \neq i}^r \rho B_j \sigma_i \sigma_j + B_i \sigma_i^2 & B^t \Lambda B + \sigma_\epsilon^2 \end{pmatrix} \end{aligned}$$

to give the distribution as required. ■

**Theorem 4.3** *The conditional distribution of  $Z_i|Y = y$  is*

$$Z_i|Y = y \sim N \left( \frac{\sum_{j=1, j \neq i}^r B_j \rho \sigma_j \sigma_i + B_i \sigma_i^2}{B^t \Lambda B + \sigma_\epsilon^2} (y - A), \sigma_i^2 - \frac{(\sum_{j=1, j \neq i}^r B_j \rho \sigma_j \sigma_i + B_i \sigma_i^2)^2}{B^t \Lambda B + \sigma_\epsilon^2} \right)$$

**Proof**

There is a result which says that if  $V = \begin{pmatrix} V_{(1)} \\ V_{(2)} \end{pmatrix}$  where  $V_{(1)}$  is a  $q \times 1$  matrix and  $V_{(2)}$  is a  $(p - q) \times 1$  matrix and is distributed as

$$N_p \left( \mu = \begin{pmatrix} \mu_{(1)} \\ \mu_{(2)} \end{pmatrix}, \Sigma = \begin{pmatrix} \Sigma_{(1,1)} & \Sigma_{(1,2)} \\ \Sigma_{(2,1)} & \Sigma_{(2,2)} \end{pmatrix} \right)$$

then the conditional distribution of  $V_{(1)}$  given  $V_{(2)} = v$  is

$$N_q(\mu_{(1)} + \Sigma_{(1,2)}\Sigma_{(2,2)}^{-1}(v - \mu_{(2)}), \Sigma_{(1,1)} - \Sigma_{(1,2)}\Sigma_{(2,2)}^{-1}\Sigma_{(2,1)}) \quad [38]$$

The distribution of  $Z_i$  and  $Y$  is

$$\begin{pmatrix} Z_i \\ Y \end{pmatrix} \sim N_2 \left( \begin{pmatrix} 0 \\ A \end{pmatrix}, \begin{pmatrix} \sigma_i^2 & \sum_{j=1, j \neq i}^n B_j \rho \sigma_j \sigma_i + B_i \sigma_i^2 \\ \sum_{j=1, j \neq i}^n B_j \rho \sigma_j \sigma_i + B_i \sigma_i^2 & B^t \Lambda B + \sigma_\epsilon^2 \end{pmatrix} \right)$$

Putting  $\mu_{(1)} = 0$ ,  $\mu_{(2)} = A$ ,  $\Sigma_{(1,1)} = \sigma_i^2$ ,  $\Sigma_{(1,2)} = \Sigma_{(2,1)} = \sum_{j=1, j \neq i}^n B_j \rho \sigma_j \sigma_i + B_i \sigma_i^2$  and  $\Sigma_{(2,2)} = B^t \Lambda B + \sigma_\epsilon^2$  gives the mean of  $Z_i|Y = y$  as

$$\mu_{(1)} + \Sigma_{(1,2)}\Sigma_{(2,2)}^{-1}(v - \mu_{(2)}) = 0 + \frac{\sum_{j=1, j \neq i}^n B_j \rho \sigma_j \sigma_i + B_i \sigma_i^2}{B^t \Lambda B + \sigma_\epsilon^2} (y - A)$$

and the variance of  $Z_i|Y = y$  as

$$\Sigma_{(1,1)} - \Sigma_{(1,2)}\Sigma_{(2,2)}^{-1}\Sigma_{(2,1)} = \sigma_i^2 - \frac{(\sum_{j=1, j \neq i}^n B_j \rho \sigma_j \sigma_i + B_i \sigma_i^2)^2}{B^t \Lambda B + \sigma_\epsilon^2}$$

to give the distribution as required. ■

## 4.2 $\epsilon|Y = y$

Ultimately the distribution of  $\epsilon$  given a value,  $y$ , of the quality control variable  $Y$  will be found. This will be done in three steps

- 1) the joint distribution of  $\epsilon$ , a partition of  $Z$  and  $Y$  will be found,
  - 2) the marginal distribution of  $\epsilon$  and  $Y$  will be found,
- and finally
- 3) the conditional distribution of  $\epsilon|Y = y$  will be found.

Now  $\mathbf{Z}$  can be partitioned into two components  $\begin{pmatrix} \mathbf{Z}_{(1)} \\ \mathbf{Z}_{(2)} \end{pmatrix}$  where  $\mathbf{Z}_{(1)} = Z_1$  and

$\mathbf{Z}_{(2)} = \begin{pmatrix} Z_2 \\ \vdots \\ Z_r \end{pmatrix}$  so that the joint distribution of  $\mathbf{Z}$  and  $Y$  can be written as

$$\begin{pmatrix} \mathbf{Z}_{(1)} \\ \mathbf{Z}_{(2)} \\ Y \end{pmatrix} \sim N_{r+1} \left( \begin{pmatrix} 0 \\ \mathbf{0}_{r-1} \\ 0 \end{pmatrix}, \begin{pmatrix} \Lambda_{(1,1)} & \Lambda_{(1,2)} & 0 \\ \Lambda_{(2,1)} & \Lambda_{(2,2)} & \mathbf{0}_{r-1} \\ 0 & \mathbf{0}_{r-1}^t & \sigma_\epsilon^2 \end{pmatrix} \right)$$

Corresponding to the partition of  $\mathbf{Z}$ ,  $\mathbf{B}$  can be partitioned into  $\mathbf{B} = \begin{pmatrix} \mathbf{B}_{(1)} \\ \mathbf{B}_{(2)} \end{pmatrix}$

where  $\mathbf{B}_{(1)} = B_1$  and  $\mathbf{B}_{(2)} = \begin{pmatrix} B_2 \\ \vdots \\ B_r \end{pmatrix}$  and  $\Lambda$  can be partitioned into

$$\Lambda = \begin{pmatrix} \Lambda_{(1,1)} & \Lambda_{(1,2)} \\ \Lambda_{(2,1)} & \Lambda_{(2,2)} \end{pmatrix}$$

where

$$\begin{aligned} \Lambda_{(1,1)} &= \Lambda_{1,1} \\ \Lambda_{(1,2)} &= \begin{pmatrix} \Lambda_{1,2} & \dots & \Lambda_{1,r} \end{pmatrix} \\ \Lambda_{(2,1)} &= \begin{pmatrix} \Lambda_{2,1} \\ \vdots \\ \Lambda_{r,1} \end{pmatrix} \end{aligned}$$

and

$$\Lambda_{(2,2)} = \begin{pmatrix} \Lambda_{2,2} & \dots & \Lambda_{2,r} \\ \vdots & \vdots & \vdots \\ \Lambda_{r,2} & \dots & \Lambda_{r,r} \end{pmatrix}$$

**Theorem 4.4** *The joint distribution of  $Y$ ,  $\mathbf{Z}_{(2)}$  and  $\epsilon$  is*

$$\begin{pmatrix} Y \\ \mathbf{Z}_{(2)} \\ \epsilon \end{pmatrix} \sim N_{r+1} \left( \begin{pmatrix} A \\ \mathbf{0}_{r-1} \\ 0 \end{pmatrix}, \begin{pmatrix} \mathbf{B}^t \Lambda \mathbf{B} + \sigma_\epsilon^2 & \mathbf{B}_{(1)} \Lambda_{(1,2)} + \mathbf{B}_{(2)}^t \Lambda_{(2,2)} & \sigma_\epsilon^2 \\ \Lambda_{(2,1)} \mathbf{B}_{(1)} + \Lambda_{(2,2)} \mathbf{B}_{(2)} & \Lambda_{(2,2)} & \mathbf{0}_{r-1} \\ 0 & \mathbf{0}_{r-1}^t & \sigma_\epsilon^2 \end{pmatrix} \right)$$

**Proof**

A transformation will be used to get the joint distribution of  $Y - A$ ,  $\mathbf{Z}_{(2)}$  and  $\epsilon$

from the joint distribution of  $Z$  and  $\epsilon$ .

There is a result which says that if  $V$  is  $N_n(\mu, \Sigma)$  and  $W = CV$  where  $C$  is an matrix of constants that has dimensions  $n \times n$  and  $\text{rank}=n$  then  $W$  is

$$N_n(C\mu, C\Sigma C^t). \quad [38]$$

Now

$$\begin{pmatrix} Y - A \\ Z_{(2)} \\ \epsilon \end{pmatrix} = \begin{pmatrix} B_{(1)} & B_{(2)}^t & 1 \\ 0_{r-1} & I_{r-1} & 0_{r-1} \\ 0 & 0_{r-1}^t & 1 \end{pmatrix} \begin{pmatrix} Z_{(1)} \\ Z_{(2)} \\ \epsilon \end{pmatrix}$$

So the transformation matrix is

$$C = \begin{pmatrix} B_{(1)} & B_{(2)}^t & 1 \\ 0_{r-1} & I_{r-1} & 0_{r-1} \\ 0 & 0_{r-1}^t & 1 \end{pmatrix}$$

Consider the matrix containing the last  $r$  rows and  $r$  columns of  $C$ . This matrix is an identity matrix of size  $r$  and hence is of rank  $r$ . Let  $b_i$  be the  $i^{th}$  element of  $B$  and there are  $r$  such elements. Perform these row operations on  $C$

$$\begin{aligned} \text{row } 1 &= \text{row } 1 - \frac{1}{b_2} \times \text{row } 2 \\ &\vdots \\ \text{row } 1 &= \text{row } 1 - \frac{1}{b_i} \times \text{row } i \\ &\vdots \\ \text{row } 1 &= \text{row } 1 - \frac{1}{b_r} \times \text{row } r \end{aligned}$$

and finally let

$$\begin{aligned} \text{row } 1 &= \text{row } 1 - \text{row } (r + 1) \\ \text{row } 1 &= \text{row } 1 \times \frac{1}{b_1} \end{aligned}$$

Now the row reduced version of  $C$ ,  $C_{\text{reduced}}$ , is an identity matrix of size  $r + 1$  and so has  $\text{rank}(C_{\text{reduced}})=r + 1$  and thus  $C$  is of full rank.

The mean of  $\begin{pmatrix} Z_{(1)} \\ Z_{(2)} \\ Y - A \end{pmatrix}$  is

$$C\mu = \begin{pmatrix} B_{(1)} & B_{(2)}^t & 1 \\ 0_{r-1} & I_{r-1} & 0_{r-1} \\ 0 & 0_{r-1}^t & 1 \end{pmatrix} \begin{pmatrix} 0 \\ 0_{r-1} \\ 0 \end{pmatrix} = \begin{pmatrix} \bullet \\ 0_{r-1} \\ \bullet \end{pmatrix}$$



and the variance is

$$\begin{aligned}
 & C\Sigma C^t \\
 = & \begin{pmatrix} B_{(1)} & B_{(2)}^t & 1 \\ 0_{r-1} & I_{r-1} & 0_{r-1} \\ 0 & 0_{r-1}^t & 1 \end{pmatrix} \begin{pmatrix} \Lambda_{(1,1)} & \Lambda_{(1,2)} & 0 \\ \Lambda_{(2,1)} & \Lambda_{(2,2)} & 0_{r-1} \\ 0 & 0_{r-1} & \sigma_\epsilon^2 \end{pmatrix} \begin{pmatrix} B_{(1)} & B_{(2)}^t & 1 \\ 0_{r-1} & I_{r-1} & 0_{r-1} \\ 0 & 0_{r-1}^t & 1 \end{pmatrix}^t \\
 = & \begin{pmatrix} B_{(1)}\Lambda_{(1,1)} + B_{(2)}^t\Lambda_{(2,1)} & B_{(1)}\Lambda_{(1,2)} + B_{(2)}^t\Lambda_{(2,2)} & \sigma_\epsilon^2 \\ \Lambda_{(2,1)} & \Lambda_{(2,2)} & 0_{r-1} \\ 0 & 0_{r-1}^t & \sigma_\epsilon^2 \end{pmatrix} \begin{pmatrix} B_{(1)} & 0_{r-1}^t & 0 \\ B_{(2)} & I_{r-1} & 0_{r-1} \\ 1 & 0_{r-1}^t & 1 \end{pmatrix} \\
 = & \begin{pmatrix} B_{(1)}\Lambda_{(1,1)}B_{(1)} + B_{(2)}^t\Lambda_{(2,1)}B_{(1)} & B_{(1)}\Lambda_{(1,2)} + B_{(2)}^t\Lambda_{(2,2)} & \sigma_\epsilon^2 \\ + B_{(1)}\Lambda_{(1,2)}B_{(2)} + B_{(2)}^t\Lambda_{(2,2)}B_{(2)} + \sigma_\epsilon^2 & \Lambda_{(2,1)}B_{(1)} + \Lambda_{(2,2)}B_{(2)} & 0_{r-1} \\ \Lambda_{(2,1)}B_{(1)} + \Lambda_{(2,2)}B_{(2)} & \Lambda_{(2,2)} & 0_{r-1} \\ & & \sigma_\epsilon^2 \end{pmatrix}
 \end{aligned}$$

hence

$$\begin{pmatrix} Y - A \\ Z_{(2)} \\ \epsilon \end{pmatrix} \sim N_{r+1} \left( \begin{pmatrix} 0 \\ 0_{r-1} \\ 0 \end{pmatrix}, \begin{pmatrix} B^t\Lambda B + \sigma_\epsilon^2 & B_{(1)}\Lambda_{(1,2)} + B_{(2)}^t\Lambda_{(2,2)} & \sigma_\epsilon^2 \\ \Lambda_{(2,1)}B_{(1)} + \Lambda_{(2,2)}B_{(2)} & \Lambda_{(2,2)} & 0_{r-1} \\ \sigma_\epsilon^2 & 0_{r-1}^t & \sigma_\epsilon^2 \end{pmatrix} \right)$$

Now

$$\begin{pmatrix} Y \\ Z_{(2)} \\ \epsilon \end{pmatrix} = \begin{pmatrix} Y - A \\ Z_{(2)} \\ \epsilon \end{pmatrix} + \begin{pmatrix} A \\ 0_{r-1} \\ 0 \end{pmatrix}$$

so

$$\begin{pmatrix} Y \\ Z_{(2)} \\ \epsilon \end{pmatrix} \sim N_{r+1} \left( \begin{pmatrix} A \\ 0_{r-1} \\ 0 \end{pmatrix}, \begin{pmatrix} B^t\Lambda B + \sigma_\epsilon^2 & B_{(1)}\Lambda_{(1,2)} + B_{(2)}^t\Lambda_{(2,2)} & \sigma_\epsilon^2 \\ \Lambda_{(2,1)}B_{(1)} + \Lambda_{(2,2)}B_{(2)} & \Lambda_{(2,2)} & 0_{r-1} \\ \sigma_\epsilon^2 & 0_{r-1}^t & \sigma_\epsilon^2 \end{pmatrix} \right)$$

■

**Theorem 4.5** *The marginal distribution of  $Y$  and  $\epsilon$  is*

$$\begin{pmatrix} Y \\ \epsilon \end{pmatrix} \sim N_2 \left( \begin{pmatrix} A \\ 0 \end{pmatrix}, \begin{pmatrix} B^t\Lambda B + \sigma_\epsilon^2 & \sigma_\epsilon^2 \\ \sigma_\epsilon^2 & \sigma_\epsilon^2 \end{pmatrix} \right)$$

**Proof**

There is a result that says if  $V$  is  $N_p(\mu, \Sigma)$  and  $C$  is a  $q \times p$  matrix of constants with  $\text{rank}(C)=q(\leq p)$  then if  $W = CV$  then  $W$  is  $N_q(C\mu, C\Sigma C^t)$ . [38]

Let  $C$  be a  $2 \times (r + 1)$  matrix where

$$C[j, k] = \begin{cases} 1 & j = 1, k = 1, k \neq r + 1 \\ 1 & j = 2, k = r + 1 \\ 0 & \text{otherwise} \end{cases}$$

So

$$C \begin{pmatrix} Y \\ Z_{(2)} \\ \epsilon \end{pmatrix} = \begin{pmatrix} 1 & 0 & \dots & 0 & 0 \\ 0 & 0 & \dots & 0 & 1 \end{pmatrix} \begin{pmatrix} Y \\ Z_{(2)} \\ \epsilon \end{pmatrix} = \begin{pmatrix} Y \\ \epsilon \end{pmatrix}$$

The  $C$  matrix has rank 2 and picks the  $1^{rst}$  and  $(r + 1)^{th}$  rows of  $\begin{pmatrix} Y \\ Z_{(2)} \\ \epsilon \end{pmatrix}$  to give

the marginal distribution of  $Y$  and  $\epsilon$ .

The mean of  $Y$  and  $\epsilon$  is

$$C\mu = \begin{pmatrix} 1 & 0 & \dots & 0 & 0 \\ 0 & 0 & \dots & 0 & 1 \end{pmatrix} \begin{pmatrix} A \\ 0 \end{pmatrix} = \begin{pmatrix} A \\ 0 \end{pmatrix}$$

and the variance is

$$\begin{aligned} & C\Lambda C^t \\ &= \begin{pmatrix} 1 & 0 & \dots & 0 & 0 \\ 0 & 0 & \dots & 0 & 1 \end{pmatrix} \begin{pmatrix} B^t\Lambda B + \sigma_\epsilon^2 & & & & \\ \Lambda_{(2,1)}B_{(1)} + \Lambda_{(2,2)}B_{(2)} & B_{(1)}\Lambda_{(1,2)} + B_{(2)}\Lambda_{(2,2)} & & \sigma_\epsilon^2 & \\ & \Lambda_{(2,2)} & & 0_{r-1} & \\ & & & \sigma_\epsilon^2 & \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \\ 0 & 1 \end{pmatrix} \\ &= \begin{pmatrix} B^t\Lambda B + \sigma_\epsilon^2 & \sigma_\epsilon^2 \\ \sigma_\epsilon^2 & \sigma_\epsilon^2 \end{pmatrix} \end{aligned}$$

to give the distribution as required. ■

**Theorem 4.6** *The conditional distributon of  $\epsilon|Y = y$  is*

$$\epsilon|Y = y \sim N\left(\frac{\sigma_\epsilon^2}{B^t\Lambda B + \sigma_\epsilon^2}(y - A), \frac{\sigma_\epsilon^2 B^t\Lambda B}{B^t\Lambda B + \sigma_\epsilon^2}\right)$$

**Proof**

There is a result which says that if  $V$  is  $N_p(\mu, \Sigma)$  then the conditional distribution of  $V_{(1)}$  given  $V_{(2)} = v$  is

$$N_q(\mu_{(1)} + \Sigma_{12}\Sigma_{(2,2)}^{-1}(v - \mu_{(2)}), \Sigma_{(1,1)} - \Sigma_{(1,2)}\Sigma_{(2,2)}^{-1}\Sigma_{(2,1)}).$$

The distribution of  $Y$  and  $\epsilon$  is

$$\begin{pmatrix} Y \\ \epsilon \end{pmatrix} \sim N_2\left(\begin{pmatrix} A \\ 0 \end{pmatrix}, \begin{pmatrix} B^t\Lambda B + \sigma_\epsilon^2 & \sigma_\epsilon^2 \\ \sigma_\epsilon^2 & \sigma_\epsilon^2 \end{pmatrix}\right)$$

This gives  $\mu_{(1)} = 0$ ,  $\mu_{(2)} = A$ ,  $\Sigma_{(1,1)} = \sigma_\epsilon^2$ ,  $\Sigma_{(1,2)} = \Sigma_{(2,1)} = \sigma_\epsilon^2$  and  $\Sigma_{(2,2)} = B^t\Lambda B + \sigma_\epsilon^2$ .

The mean of  $\epsilon|Y = y$  is

$$\begin{aligned} \mu_{(1)} + \Sigma_{(1,2)}\Sigma_{(2,2)}^{-1}(v - \mu_{(2)}) &= 0 + \sigma_\epsilon^2 \times (B^t\Lambda B + \sigma_\epsilon^2)^{-1}(y - A) \\ &= \frac{\sigma_\epsilon^2(y - A)}{B^t\Lambda B + \sigma_\epsilon^2} \end{aligned}$$

The variance of  $\epsilon|Y = y$  is

$$\begin{aligned}\Sigma_{(1,1)} - \Sigma_{(1,2)}\Sigma_{(2,2)}^{-1}\Sigma_{(2,1)} &= \sigma_\epsilon^2 - \sigma_\epsilon^2 \times \sigma_\epsilon^2 \times (B^t\Lambda B + \sigma_\epsilon^2)^{-1} \\ &= \sigma_\epsilon^2 - \frac{\sigma_\epsilon^4}{B^t\Lambda B + \sigma_\epsilon^2} \\ &= \frac{\sigma_\epsilon^2 B^t\Lambda B}{B^t\Lambda B + \sigma_\epsilon^2}\end{aligned}$$

which gives the distribution of  $\epsilon|Y = y$  as required. ■

Therefore the mean of  $Z_i|Y = Y$  can be used as an estimate of  $Z_i$ . Similarly the mean of  $\epsilon|Y = y$  can be used as an estimate of  $\epsilon$ .

### 4.3 Highest Probability Estimates

An alternative method of predicting  $Z$  and  $\epsilon$  is by using calculus with Lagrange multipliers. In this method the most likely values of  $Z$  and  $\epsilon$  will be found subject to the condition that  $Y = A + Z^t B + \epsilon$ , the regression equation, found through experimental design.

Recall the model that was introduced in the previous section with the relevant information being listed below. The unobservable, uncontrollable variables,  $Z$ , are distributed  $N_r(0_r, \Lambda_2)$  and  $\Lambda_2$  will be written as  $\Lambda$  for simplicity. The pure random error,  $\epsilon$ , is distributed as  $N(0, \sigma_\epsilon^2)$ . Let  $\epsilon$  and  $Z$  be independent and have joint distribution

$$f_{Z,\epsilon}(z, \epsilon) = d \exp \left[ -\frac{1}{2} \left( z^t \Lambda^{-1} z + \frac{\epsilon^2}{\sigma_\epsilon^2} \right) \right]$$

where  $d$  is a constant required to make  $f_{Z,\epsilon}(z, \epsilon)$  a probability density function. The quality variable,  $Y$ , is related to  $Z$  and  $\epsilon$  by the equation

$$Y = A + Z^t B + \epsilon$$

Finding the most likely value of  $Z$  and  $\epsilon$  for our value of  $Y = y$  will involve maximising  $f_{Z,\epsilon}(z, \epsilon)$  subject to  $Y = A + Z^t B + \epsilon$ .

**Theorem 4.7** *If the Variance-Covariance matrix of the joint distribution of  $Z$  and  $\epsilon$  is positive definite then the values of  $Z$  and  $\epsilon$  that maximise their joint*

distribution

$$f_{\mathbf{z},\epsilon}(\mathbf{z}, \epsilon) = d \exp \left[ -\frac{1}{2} \left( \mathbf{z}^t \boldsymbol{\Lambda}^{-1} \mathbf{z} + \frac{\epsilon^2}{\sigma^2} \right) \right]$$

subject to

$$y = A + \mathbf{z}^t \mathbf{B} + \epsilon$$

are

$$\hat{\mathbf{z}} = \frac{\boldsymbol{\Lambda} \mathbf{B}}{\mathbf{B}^t \boldsymbol{\Lambda} \mathbf{B} + \sigma^2} (y - A)$$

and

$$\hat{\epsilon} = \frac{\sigma^2}{\mathbf{B}^t \boldsymbol{\Lambda} \mathbf{B} + \sigma^2} (y - A)$$

### Proof

The joint distribution of  $\mathbf{Z}$  and  $\epsilon$  is

$$f_{\mathbf{z},\epsilon}(\mathbf{z}, \epsilon) = d \exp \left[ -\frac{1}{2} \left( \mathbf{z}^t \boldsymbol{\Lambda}^{-1} \mathbf{z} + \frac{\epsilon^2}{\sigma^2} \right) \right]$$

which is to be maximised subject to the condition that

$$y = A + \mathbf{z}^t \mathbf{B} + \epsilon$$

This is equivalent to maximising

$$-\frac{1}{2} \left( \mathbf{z}^t \boldsymbol{\Lambda} \mathbf{z} + \frac{\epsilon^2}{\sigma^2} \right)$$

subject to

$$y = A + \mathbf{z}^t \mathbf{B} + \epsilon$$

Let

$$L = -\frac{1}{2} \mathbf{z}^t \boldsymbol{\Lambda}^{-1} \mathbf{z} - \frac{1}{2} \frac{\epsilon^2}{\sigma^2} + \lambda (y - A - \mathbf{z}^t \mathbf{B} - \epsilon)$$

Now

$$\frac{\partial L}{\partial \mathbf{z}} = -\boldsymbol{\Lambda}^{-1} \mathbf{z} - \lambda \mathbf{B}$$

Setting this equal to zero gives

$$\mathbf{z} = -\lambda \boldsymbol{\Lambda} \mathbf{B}$$

Also

$$\frac{\partial L}{\partial \epsilon} = -\frac{\epsilon}{\sigma^2} - \lambda$$

Setting this equal to zero gives

$$\epsilon = -\lambda \sigma^2$$

And

$$\begin{aligned}\frac{\partial L}{\partial \lambda} &= y - A - \mathbf{z}^t \mathbf{B} - \epsilon \\ &= y - A + \lambda \mathbf{B}^t \mathbf{\Lambda}^t \mathbf{B} + \lambda \sigma^2\end{aligned}$$

Setting this equal to zero gives

$$\lambda = -\frac{y - A}{\mathbf{B}^t \mathbf{\Lambda}^t \mathbf{B} + \sigma^2}$$

Therefore

$$\hat{\mathbf{z}} = \frac{\mathbf{\Lambda} \mathbf{B}}{\mathbf{B}^t \mathbf{\Lambda} \mathbf{B} + \sigma^2} (y - A)$$

and

$$\hat{\epsilon} = \frac{\sigma^2}{\mathbf{B}^t \mathbf{\Lambda} \mathbf{B} + \sigma^2} (y - A)$$

The final requirement is to check that these solutions give a maximum. The matrix of second derivatives is

$$\begin{pmatrix} \frac{\partial^2 L}{\partial \mathbf{z} \partial \mathbf{z}^t} = -\mathbf{\Lambda}^{-1} & \frac{\partial^2 L}{\partial \mathbf{z} \partial \epsilon} = \mathbf{0}_r \\ \frac{\partial^2 L}{\partial \epsilon \partial \mathbf{z}^t} = \mathbf{0}_r^t & \frac{\partial^2 L}{\partial \epsilon^2} = -\sigma^{-2} \end{pmatrix} = - \begin{pmatrix} \mathbf{\Lambda}^{-1} & \mathbf{0}_r \\ \mathbf{0}_r^t & \sigma^{-2} \end{pmatrix}$$

This matrix is the negative inverse of the Variance-Covariance matrix of the joint distribution of  $\mathbf{z}$  and  $\epsilon$ . Given that the Variance-Covariance is positive definite then its inverse is positive definite. The negative of a positive definite matrix gives a negative definite matrix. Hence the solutions for  $\mathbf{z}$  and  $\epsilon$  give a maximum. ■

Note that these solutions match the solutions found by finding the means of the distributions of  $\mathbf{Z}|Y = y$  and  $\epsilon|Y = y$ .

### Example

Consider the regression equation

$$y = 10 + 6z_1 + 7z_2 + \epsilon$$

with  $z_1$  and  $z_2$  having a multi-variate Normal distribution, with mean zero, and with a Variance-Covariance matrix of  $\mathbf{\Lambda}$ . Let  $\epsilon$  have a standard Normal distribution and be independent of  $z_1$  and  $z_2$  and have a joint multi-variate Normal distribution. 10,000 random multi-variate random Normal variates were

generated for  $z_1$ ,  $z_2$  and  $\epsilon$  with mean zero and for several different Variance-Covariance matrices as outlined in Table 4.1.

The specific Variance-Covariance matrices that will be of interest in this example are those that have this form

$$\begin{aligned} \Lambda &= \begin{pmatrix} \sigma_{z_1}^2 & \sigma_{z_1}\sigma_{z_2}\rho & 0 \\ \sigma_{z_1}\sigma_{z_2}\rho & \sigma_{z_2}^2 & 0 \\ 0 & 0 & 1 \end{pmatrix} \\ &= \begin{pmatrix} 1 & \rho & 0 \\ \rho & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad \begin{array}{l} \text{i.e. in this example} \\ \sigma_{z_1} = \sigma_{z_2} = 1 \end{array} \end{aligned}$$

where  $\rho$  is the correlation coefficient between  $z_1$  and  $z_2$ .

Theorem 4.7 relies on this matrix being positive definite. A matrix will be positive definite if there is a non-singular matrix  $C$  such that  $\Lambda = CC^t$ . Using Maple the following matrices were constructed to find  $C$ .

```
>P:=evalm(matrix(3,3,[0,1/(2^(1/2)),1/(2^(1/2))
,0,1/(2^(1/2)),1/(2^(1/2)),1,0,0]));
      [          1/2          1/2 ]
      [ 0 - 1/2 2          1/2 2   ]
      [          ]
P := [          1/2          1/2 ]
      [ 0  1/2 2          1/2 2   ]
      [          ]
      [ 1          0          0   ]
```

```
> Ahalf:=evalm(matrix(3,3,[1,0,0,0,(1-rho)^(1/2)
,0,0,0,(1+rho)^(1/2)]));
      [ 1          0          0   ]
      [          ]
      [          1/2          ]
Ahalf := [ 0 (1 - rho)          0   ]
      [          ]
      [          1/2          ]
      [ 0          0          (1 + rho) ]
```

```
> C:=evalm(P&*Ahalf);
```

```

      [          1/2          1/2          1/2          1/2 ]
      [ 0 - 1/2 2 (1 - rho)  1/2 2 (1 + rho) ]
      [
C := [          1/2          1/2          1/2          1/2 ]
      [ 0  1/2 2 (1 - rho)  1/2 2 (1 + rho) ]
      [
      [ 1          0          0          0 ]

```

```
> evalm(C&*transpose(C));
```

```

      [ 1  rho  0 ]
      [          ]
      [ rho  1  0 ]
      [          ]
      [ 0  0  1 ]

```

So there is a matrix  $C$ , namely  $C = P \times \mathbf{Ahalf}$ , such that  $\Lambda = CC^t$ .  $C$  will be non-singular as long as  $\rho$  doesn't equal  $\pm 1$ . Therefore  $\Lambda$  is positive definite and the theorem will hold for all values of  $\rho$  except  $\pm 1$ .

10,000 estimates for  $z_1$ ,  $z_2$  and  $\epsilon$  were created using the estimators from the Lagrange method. These were compared with  $z_1$ ,  $z_2$  and  $\epsilon$  and an estimated MSE for each Variance-Covariance combination and estimator was calculated. The results are contained in Table 4.1.

Table 4.1: This table gives the sample Mean Square Error for each estimate of  $z_1$ ,  $z_2$  and  $\epsilon$  for different Variance-Covariance matrices of the joint distribution of  $z_1$ ,  $z_2$  and  $\epsilon$ .

Variance-covariance Matrix	Total Variance $B^t \Lambda B + \sigma_\epsilon^2$	Variables	Estimated MSE $E(\text{predicted} - \text{actual})^2$
$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$	85	$z_1$ $z_2$ $\epsilon$	0.5852 0.4356 0.9891
$\begin{pmatrix} 1 & 0.75 & 0 \\ 0.75 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$	148	$z_1$ $z_2$ $\epsilon$	0.1504 0.1121 0.9925
$\begin{pmatrix} 1 & -0.75 & 0 \\ -0.75 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$	22	$z_1$ $z_2$ $\epsilon$	0.9775 0.7288 0.9546

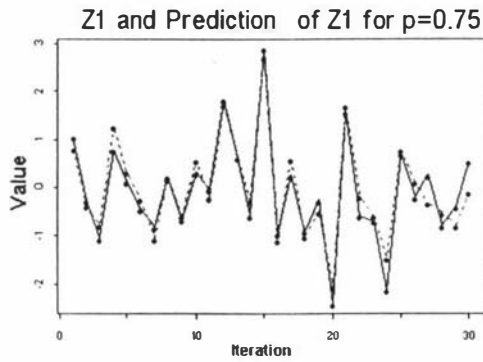
The figures that follow give a visual indication of how well the estimators of  $z$  and  $\epsilon$  do for different values of the Variance-Covariance matrix.

### Case 1.

The Figures 4.1, 4.2 and 4.3 give time series plots of the first 30 values of  $z_1$ ,  $z_2$  and  $\epsilon$  and their estimates when the Variance-Covariance matrix is

$$\Lambda = \begin{pmatrix} 1 & 0.75 & 0 \\ 0.75 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

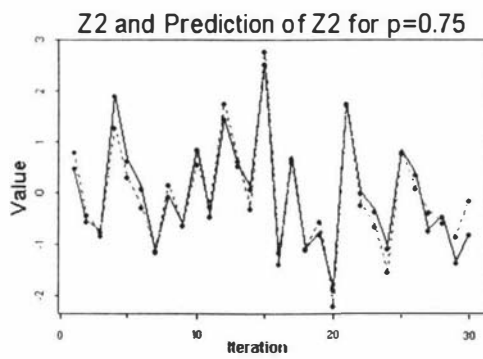




The  $z_1$  values are joined by solid lines.

The  $\hat{z}_1$  values are joined by broken lines.

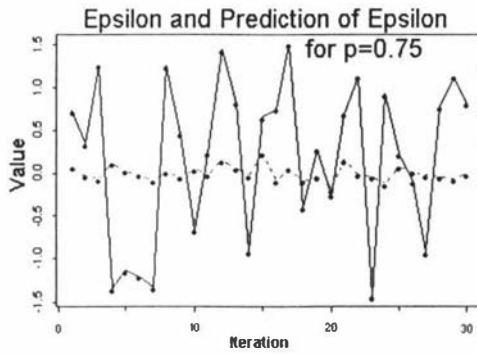
Figure 4.1: The graph of the value of  $z_1$  and the predictions of  $z_1$  when the covariance (and correlation) between  $z_1$  and  $z_2$  is 0.75.



The  $z_2$  values are joined by solid lines.

The  $\hat{z}_2$  values are joined by broken lines.

Figure 4.2: The graph of the value of  $z_2$  and the predictions of  $z_2$  when the covariance (and correlation) between  $z_1$  and  $z_2$  is 0.75.



The  $\epsilon$  values are joined by solid lines.

The  $\hat{\epsilon}$  values are joined by broken lines.

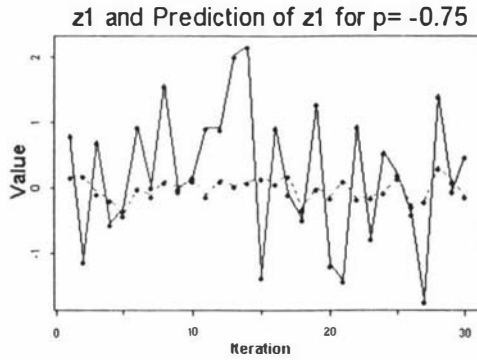
Figure 4.3: The graph of the value of  $\epsilon$  and the predictions of  $\epsilon$  when the covariance (and correlation) between  $z_1$  and  $z_2$  is 0.75.

Clearly the estimators of  $z_1$  and  $z_2$  are doing a good job while the estimator of  $\epsilon$  is doing a poor job.

### Case 2.

The Figures 4.4, 4.5 and 4.6 give time series plots of the first 30 values of  $z_1$ ,  $z_2$  and  $\epsilon$  and their estimates when the Variance-Covariance matrix is

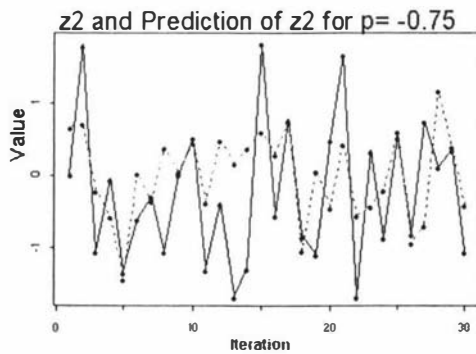
$$\mathbf{\Lambda} = \begin{pmatrix} 1 & -0.75 & 0 \\ -0.75 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$



The  $z_1$  values are joined by solid lines.

The  $\hat{z}_1$  values are joined by broken lines.

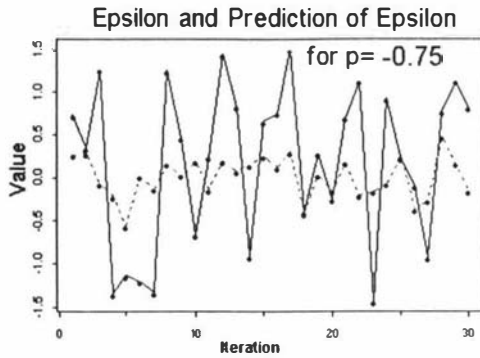
Figure 4.4: The graph of the value of  $z_1$  and the predictions of  $z_1$  when the covariance (and correlation) between  $z_1$  and  $z_2$  is  $-0.75$ .



The  $z_2$  values are joined by solid lines.

The  $\hat{z}_2$  values are joined by broken lines.

Figure 4.5: The graph of the value of  $z_2$  and the predictions of  $z_2$  when the covariance (and correlation) between  $z_1$  and  $z_2$  is  $-0.75$ .



The  $\epsilon$  values are joined by solid lines.  
 The  $\hat{\epsilon}$  values are joined by broken lines.

Figure 4.6: The graph of the value of  $\epsilon$  and the predictions of  $\epsilon$  when the covariance (and correlation) between  $z_1$  and  $z_2$  is  $-0.75$ .

In this case the estimators for  $z_1$ ,  $z_2$  and  $\epsilon$  are doing a poor job. The performance of the estimators varies.

### 4.4 Evaluating the Performance of the Estimators

From the examples in the previous section it is obvious that the estimators  $\hat{z}$  and  $\hat{\epsilon}$  vary in their ability to estimate  $z$  and  $\epsilon$  well. Therefore a measure of how well those estimators perform would be a useful tool. A way of making a measure is to consider the Mean Square Error between  $\hat{z}$  and  $z$  and between  $\hat{\epsilon}$  and  $\epsilon$ . Recall that the estimators of  $z$  and  $\epsilon$  are

$$\hat{z} = \frac{\Lambda B}{B^t \Lambda B + \sigma^2} (y - A)$$

and

$$\hat{\epsilon} = \frac{\sigma^2}{B^t \Lambda B + \sigma^2} (y - A)$$

**Theorem 4.8** *The Mean Square Error of  $\hat{z}$  for  $z$  is*

$$E((\hat{z} - z)(\hat{z} - z)^t) = \Lambda - \frac{\Lambda B B^t \Lambda}{B^t \Lambda B + \sigma^2}$$

**Proof**

Consider

$$\begin{aligned} E((\hat{\mathbf{z}} - \mathbf{z})(\hat{\mathbf{z}} - \mathbf{z})^t) &= E((\hat{\mathbf{z}} - \mathbf{z})(\hat{\mathbf{z}}^t - \mathbf{z}^t)) \\ &= E(\hat{\mathbf{z}}\hat{\mathbf{z}}^t) - E(\hat{\mathbf{z}}\mathbf{z}^t) - E(\mathbf{z}\hat{\mathbf{z}}^t) + E(\mathbf{z}\mathbf{z}^t) \end{aligned}$$

Evaluating each expectation separately gives

1)

$$\begin{aligned} E(\hat{\mathbf{z}}\hat{\mathbf{z}}^t) &= E\left(\frac{\Lambda\mathbf{B}}{\mathbf{B}^t\Lambda\mathbf{B} + \sigma_\epsilon^2}(Y - A)(Y - A)\frac{\mathbf{B}^t\Lambda}{\mathbf{B}^t\Lambda\mathbf{B} + \sigma_\epsilon^2}\right) \\ &= \frac{\Lambda\mathbf{B}}{\mathbf{B}^t\Lambda\mathbf{B} + \sigma_\epsilon^2}E((Y - A)^2)\frac{\mathbf{B}^t\Lambda}{\mathbf{B}^t\Lambda\mathbf{B} + \sigma_\epsilon^2} \\ &= \frac{\Lambda\mathbf{B}}{\mathbf{B}^t\Lambda\mathbf{B} + \sigma_\epsilon^2}(\mathbf{B}^t\Lambda\mathbf{B} + \sigma_\epsilon^2)\frac{\mathbf{B}^t\Lambda}{\mathbf{B}^t\Lambda\mathbf{B} + \sigma_\epsilon^2} \\ &= \frac{\Lambda\mathbf{B}\mathbf{B}^t\Lambda}{\mathbf{B}^t\Lambda\mathbf{B} + \sigma_\epsilon^2} \end{aligned}$$

since

$$\begin{aligned} E((Y - A)^2) &= E((\mathbf{B}^t\mathbf{z} + \epsilon)^2) \\ &= \text{Var}(\mathbf{B}^t\mathbf{z}) + \text{Var}(\epsilon) \\ &\quad \text{since } E(\mathbf{B}^t\mathbf{z}) = 0, E(\epsilon) = 0 \text{ and } \mathbf{Z} \text{ is independent of } \epsilon \\ &= \mathbf{B}^t\Lambda\mathbf{B} + \sigma_\epsilon^2 \end{aligned}$$

2)

$$\begin{aligned} E(\hat{\mathbf{z}}\mathbf{z}^t) &= E\left(\frac{\Lambda\mathbf{B}}{\mathbf{B}^t\Lambda\mathbf{B} + \sigma_\epsilon^2}(Y - A)\mathbf{z}^t\right) \\ &= \frac{\Lambda\mathbf{B}}{\mathbf{B}^t\Lambda\mathbf{B} + \sigma_\epsilon^2}E((\mathbf{B}^t\mathbf{z} + \epsilon)\mathbf{z}^t) \\ &= \frac{\Lambda\mathbf{B}}{\mathbf{B}^t\Lambda\mathbf{B} + \sigma_\epsilon^2}(\mathbf{B}^tE(\mathbf{z}\mathbf{z}^t) + E(\epsilon\mathbf{z}^t)) \\ &= \frac{\Lambda\mathbf{B}\mathbf{B}^t\Lambda}{\mathbf{B}^t\Lambda\mathbf{B} + \sigma_\epsilon^2} \end{aligned}$$

as

$$E(\epsilon\mathbf{z}^t) = E(\epsilon)E(\mathbf{z}^t) = \mathbf{0}_r$$

3)

$$\begin{aligned}
 E(z\hat{z}^t) &= E\left(z\left((y-A)\frac{\Lambda B}{B^t\Lambda B + \sigma_\epsilon^2}\right)^t\right) \\
 &= E\left(z(y-A)\left(\frac{B^t\Lambda}{B^t\Lambda B + \sigma_\epsilon^2}\right)^t\right) \\
 &= (E(zz^t)B + E(z\epsilon))\frac{B^t\Lambda}{B^t\Lambda B + \sigma_\epsilon^2} \\
 &= \frac{\Lambda BB^t\Lambda}{B^t\Lambda B + \sigma_\epsilon^2}
 \end{aligned}$$

4)

$$E(zz^t) = \text{var}(z) = \Lambda$$

Putting these together gives

$$\begin{aligned}
 E((\hat{z} - z)(\hat{z} - z)^t) &= E((\hat{z} - z)(\hat{z}^t - z^t)) \\
 &= E(\hat{z}\hat{z}^t) - E(\hat{z}z^t) - E(z\hat{z}^t) + E(zz^t) \\
 &= \frac{\Lambda BB^t\Lambda}{B^t\Lambda B + \sigma_\epsilon^2} - \frac{\Lambda BB^t\Lambda}{B^t\Lambda B + \sigma_\epsilon^2} - \frac{\Lambda BB^t\Lambda}{B^t\Lambda B + \sigma_\epsilon^2} + \Lambda \\
 &= \Lambda - \frac{\Lambda BB^t\Lambda}{B^t\Lambda B + \sigma_\epsilon^2}
 \end{aligned}$$

■

Now  $\Lambda - \frac{\Lambda BB^t\Lambda}{B^t\Lambda B + \sigma_\epsilon^2}$  is a square matrix with the  $i^{\text{th}}, i^{\text{th}}$  element being  $E((z_i - \hat{z}_i)^2)$  which is the Mean Square Error of  $\hat{z}_i$  for  $z_i$ .

Similarly a Mean Square Error can be generated for  $\hat{\epsilon}$ .

**Theorem 4.9** *The Mean Square Error of  $\hat{\epsilon}$  is*

$$E((\hat{\epsilon} - \epsilon)^2) = \sigma_\epsilon^2 - \frac{\sigma_\epsilon^4}{B^t\Lambda B + \sigma_\epsilon^2}$$

**Proof**

Consider

$$E((\hat{\epsilon} - \epsilon)^2) = E(\hat{\epsilon}^2) - 2E(\hat{\epsilon}\epsilon) + E(\epsilon^2)$$

Evaluating each expectation separately gives

1)

$$E(\hat{\epsilon}^2) = E\left(\left(\frac{\sigma_\epsilon^2(Y-A)}{B^t\Lambda B + \sigma_\epsilon^2}\right)^2\right)$$

$$\begin{aligned}
&= \left( \frac{\sigma_\epsilon^2}{\mathbf{B}^t \Lambda \mathbf{B} + \sigma_\epsilon^2} \right)^2 E((Y - A)^2) \\
&= \left( \frac{\sigma_\epsilon^2}{\mathbf{B}^t \Lambda \mathbf{B} + \sigma_\epsilon^2} \right)^2 E((\mathbf{B}^t \mathbf{z} + \epsilon)^2) \\
&= \left( \frac{\sigma_\epsilon^2}{\mathbf{B}^t \Lambda \mathbf{B} + \sigma_\epsilon^2} \right)^2 (Var(\mathbf{B}^t \mathbf{z}) + Var(\epsilon)) \\
&= \left( \frac{\sigma_\epsilon^2}{\mathbf{B}^t \Lambda \mathbf{B} + \sigma_\epsilon^2} \right)^2 (\mathbf{B}^t \Lambda \mathbf{B} + \sigma_\epsilon^2) \\
&= \frac{\sigma_\epsilon^4}{\mathbf{B}^t \Lambda \mathbf{B} + \sigma_\epsilon^2}
\end{aligned}$$

2)

$$\begin{aligned}
E(\hat{\epsilon}) &= E \left( \left( \frac{\sigma_\epsilon^2 (Y - A)}{\mathbf{B}^t \Lambda \mathbf{B} + \sigma_\epsilon^2} \right) \epsilon \right) \\
&= \left( \frac{\sigma_\epsilon^2}{\mathbf{B}^t \Lambda \mathbf{B} + \sigma_\epsilon^2} \right) E((\mathbf{B}^t \mathbf{z} + \epsilon)\epsilon) \\
&= \left( \frac{\sigma_\epsilon^2}{\mathbf{B}^t \Lambda \mathbf{B} + \sigma_\epsilon^2} \right) (E(\mathbf{B}^t \mathbf{z}\epsilon) + E(\epsilon^2)) \\
&= \left( \frac{\sigma_\epsilon^2}{\mathbf{B}^t \Lambda \mathbf{B} + \sigma_\epsilon^2} \right) \sigma_\epsilon^2 \\
&= \left( \frac{\sigma_\epsilon^4}{\mathbf{B}^t \Lambda \mathbf{B} + \sigma_\epsilon^2} \right)
\end{aligned}$$

3)

$$E(\epsilon^2) = Var(\epsilon) = \sigma_\epsilon^2$$

Putting these together gives

$$\begin{aligned}
E((\hat{\epsilon} - \epsilon)^2) &= E(\hat{\epsilon}^2) - 2E(\hat{\epsilon}\epsilon) + E(\epsilon^2) \\
&= \frac{\sigma_\epsilon^4}{\mathbf{B}^t \Lambda \mathbf{B} + \sigma_\epsilon^2} - 2 \left( \frac{\sigma_\epsilon^4}{\mathbf{B}^t \Lambda \mathbf{B} + \sigma_\epsilon^2} \right) + \sigma_\epsilon^2 \\
&= \sigma_\epsilon^2 - \frac{\sigma_\epsilon^4}{\mathbf{B}^t \Lambda \mathbf{B} + \sigma_\epsilon^2}
\end{aligned}$$

■

### Example Continued

Using the model and data from the previous example means that the Mean Square Error for each estimate can be generated and compared with those found from the simulation. This data is in Table 4.2.

Table 4.2: This table gives the sample Mean Square Error and theoretical Mean Square Error for each estimate of  $z_1$ ,  $z_2$  and  $\epsilon$  for different Variance-Covariance matrices of the joint distribution of  $z_1$ ,  $z_2$  and  $\epsilon$ .

Variance-covariance	Variables	Estimated MSE	MSE
$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$	$z_1$	0.5852	0.5813
	$z_2$	0.4356	0.4302
	$\epsilon$	0.9891	0.9883
$\begin{pmatrix} 1 & 0.75 & 0 \\ 0.75 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$	$z_1$	0.1504	0.1506
	$z_2$	0.1121	0.1124
	$\epsilon$	0.9925	0.9933
$\begin{pmatrix} 1 & -0.75 & 0 \\ -0.75 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$	$z_1$	0.9775	0.9755
	$z_2$	0.7288	0.7282
	$\epsilon$	0.9546	0.96

The actual values of the Mean Square Error are very close to those found through simulation.

The Mean Square Error for each variable,  $z_1$ ,  $z_2$  and  $\epsilon$ , depends on the correlation,  $\rho$ , between  $z_1$  and  $z_2$ . Using the model from the example above with different values of  $\rho$  enables the Mean Square Error for each estimate to be plotted against the correlation coefficient.

In Figure 4.7 the Mean Square Error of  $\hat{z}_1$  for  $z_1$  has been plotted against the correlation coefficient of  $z_1$  and  $z_2$ .



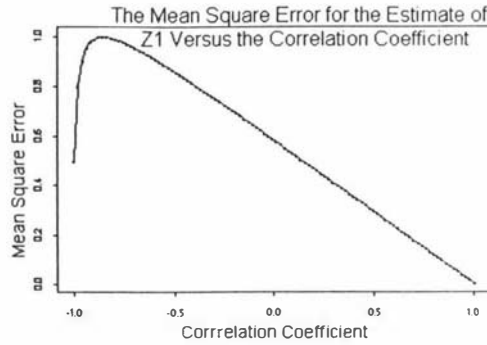


Figure 4.7: The graph of the Mean Square Error for  $\hat{z}_1$ , the estimate of  $z_1$ , versus the Correlation Coefficient of  $z_1$  and  $z_2$ .

The estimate works best when the MSE is smallest. For this case this is when  $\rho$  tends to 1 and is  $MSE = 0.006$ . The MSE peaks at  $\rho = -0.857$  with a value of the  $MSE = 1$ . In comparison estimating  $z_1$  with a random standard Normal variate would give a MSE of 2.

In Figure 4.8 the Mean Square Error of  $\hat{z}_2$  for  $z_2$  has been plotted against the correlation coefficient of  $z_1$  and  $z_2$ .

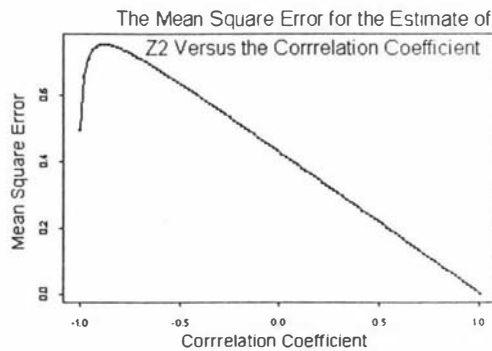


Figure 4.8: The graph of the Mean Square Error for  $\hat{z}_2$ , the estimate of  $z_2$ , versus the Correlation Coefficient of  $z_1$  and  $z_2$ .

For this case the MSE is smallest when  $\rho$  tends to 1 and is  $MSE = 0.006$ . The MSE peaks  $\rho = -0.881$  with a value of  $MSE = 0.755$ .

In Figure 4.9 the Mean Square Error of  $\hat{\epsilon}$  for  $\epsilon$  has been plotted against the correlation coefficient of  $z_1$  and  $z_2$ .

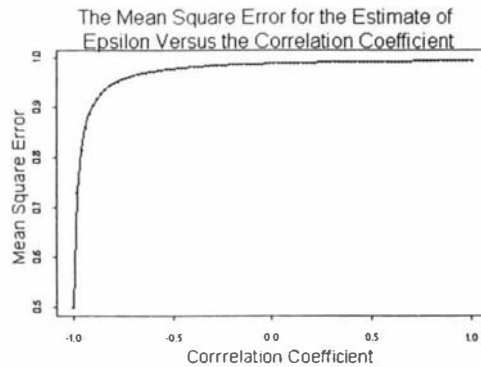


Figure 4.9: The graph of the Mean Square Error for  $\hat{\epsilon}$ , the estimate of  $\epsilon$ , versus the Correlation Coefficient of  $z_1$  and  $z_2$

For this case MSE is smallest when  $\rho$  tends to -1 and is  $MSE = 0.5$ . The MSE peaks when  $\rho$  tends to 1 with a value of  $MSE = 0.994$ . However for  $\rho \geq -0.90$  the MSE is greater than 0.9.

These estimates vary widely in their ability to estimate  $z_1$ ,  $z_2$  and  $\epsilon$  well.

## 4.5 Sensitivity to Changes in the Process

During on-line process control the quality control variable,  $Y$ , is recorded on an  $X$ -Chart. Control Limits are set at  $\mu_Y \pm 3\sigma_Y$  and any observations falling outside these limits are investigated. There are two occasions when this can occur

- 1) when the observation falls outside the control limits purely by chance or
- 2) when there has been a change in the system e.g. a change in process mean or variance.

The occurrence of an observation outside the control limits when it happens purely by chance is called a false alarm. The probability of a false alarm when the control limits are set at  $\mu_Y \pm 3\sigma_Y$  has a value of 0.0027 or 2.7 observations in every 1000. The average run length to a false alarm, ARL-to-false-alarm, is 370 observations.

It is costly to have a system that generates false alarms and so it is beneficial to have the ARL-to-false-alarm statistic as large as possible. When there has been a change in the process mean or variance then it is beneficial to have this change detected as quickly as possible. The average run length to detection,

ARL-to-detection, counts the number of observations from a change in the process until an out of control condition is found. The smaller the ARL-to-detection the better.

In order that the estimates of  $z$  and  $\epsilon$  are to be useful in detecting changes in the system they must have large values of ARL-to-false-alarm and small values of ARL-to-detection. They must also be able to point out which particular variable is the cause of the change.

### 4.5.1 Example

Consider again the example given in the previous section i.e. The regression equation is

$$y = 10 + 6z_1 + 7z_2 + \epsilon$$

with  $z_1$  and  $z_2$  having a multivariate Normal distribution with mean zero and variance  $\Lambda$ . Let  $\epsilon$  have a standard Normal distribution and be independent of  $z_1$  and  $z_2$ . Together let all three variables have a multivariate Normal distribution. One hundred thousand variates were generated for  $z_1$ ,  $z_2$  and  $\epsilon$  with mean zero and Variance-Covariance matrix

$$\begin{pmatrix} 1 & \rho & 0 \\ \rho & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

where  $\rho$  is the correlation between  $z_1$  and  $z_2$ .

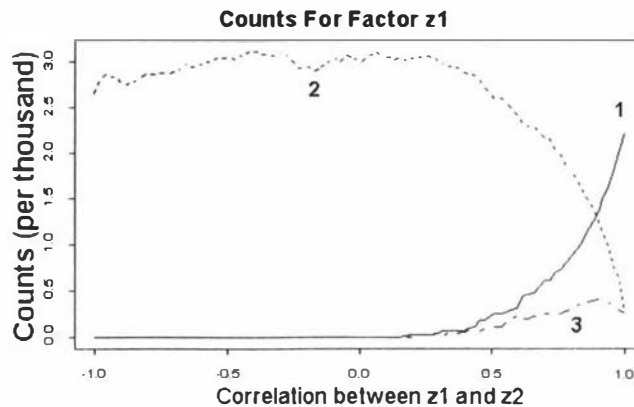
Finally  $z_1$ ,  $z_2$  and  $\epsilon$  were used to generate  $y$  which was in turn used to give the estimates  $\hat{z}_1$ ,  $\hat{z}_2$  and  $\hat{\epsilon}$ . This was repeated for different values of  $\rho$ .

These estimates will now be evaluated for their use on X-charts.

The control limits for each X-chart were set at three times the variables variance. In this case all three variables had control limits at  $\pm 3$ . Three counts were made for each estimate. A count

- 1) of when the variable and its estimate were both outside the control limits
- 2) of when the variable was outside the control limits and the estimate was inside and
- 3) of when the variable was inside the control limits and the estimate was outside.

These counts are graphed in Figures 4.10, 4.11 and 4.12 for different values of  $\rho$ , the correlation between  $z_1$  and  $z_2$ .



A count of when

- 1)  $z_1$  and  $\hat{z}_1$  were both outside the control limits
- 2)  $z_1$  was outside the control limits and  $\hat{z}_1$  was inside and
- 3)  $z_1$  was inside the control limits and  $\hat{z}_1$  was outside.

Figure 4.10: Counts of  $z_1$  and its estimate for being inside and outside the control limits.

#### 1)Line 2

When  $-1 \leq \rho < 0.25$  the estimates of  $z_1$  do not estimate  $z_1$  as well compared to when  $0.25 \leq \rho < 1$ . In the former region  $\hat{z}_1$  is underestimating  $z_1$ . This is reflected in line 2 where the count is of the number of times  $z_1$  is outside the control limits and  $\hat{z}_1$  is inside. This count is the highest of the three counts and its value is observed to be in the interval (2.5, 3) per 1000 variates. From the point of view of estimation this underestimation is poor behaviour but in terms of quality control this behaviour is welcome since it indicates a large value of ARL-to-false-alarm.

#### 2)Line 1

For line 1, which counts when  $z_1$  and  $\hat{z}_1$  are both outside the control limits, the count is remaining close to zero per 1000 variates until  $0.25 < \rho < 1$  where it increases steadily to peak at a count of 2.2 per 1000 variates. In the region  $-1 < \rho < 0.25$  there will be a very low probability of getting a false alarm but this will increase steadily in the region  $.25 < \rho < 1$ .

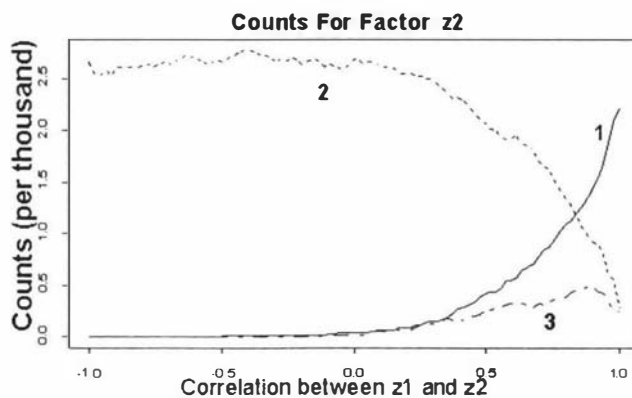
#### 3)Line 3

For line 3, which counts when  $z_1$  is within the control limits and  $\hat{z}_1$  is outside the control limits, the count is remaining close to zero per 1000 variates until  $0.25 < \rho < 1$  where it increases steadily to peak at a count of 0.4 per 1000

variates. In the region  $-1 < \rho < 0.25$  there will be a very low probability of getting a false alarm when the process is under control. In the region  $0.25 < \rho < 1$  there is a small probability of generating false alarms when the estimate for  $z_1$  is within the control limits.

The number of false alarms per 1000 variates ranges from 0 to peak at 2.6 at  $\rho = 1$ . In comparison to an X-Chart of the quality control variable  $Y$  with 2.7 false alarms per 1000 variates this offers a very good to slight improvement.

The graph for  $z_2$  is in Figure 4.11. It is very similar to  $z_1$ . The relatively minor change is that  $z_2$  is estimated comparatively better for smaller values of  $\rho$  i.e. for  $0 < \rho < 1$ .

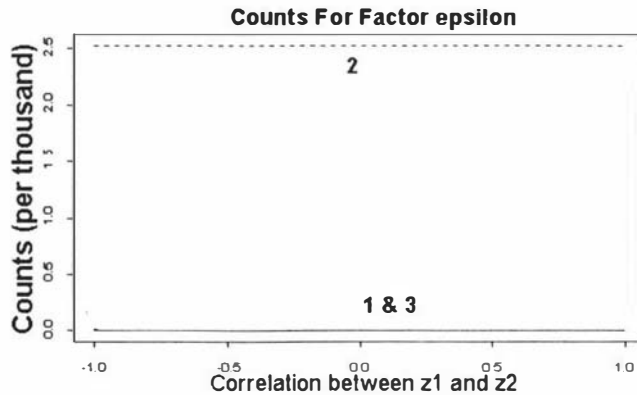


A count of when

- 1)  $z_2$  and  $\hat{z}_2$  were both outside the control limits
- 2)  $z_2$  was outside the control limits and  $\hat{z}_2$  was inside and
- 3)  $z_2$  was inside the control limits and  $\hat{z}_2$  was outside.

Figure 4.11: Counts of  $z_2$  and its estimate for being inside and outside the control limits.

A similar graph for  $\epsilon$  is in Figure 4.12.



A count of when

- 1)  $\epsilon$  and  $\hat{\epsilon}$  were both outside the control limits
- 2)  $\epsilon$  was outside the control limits and  $\hat{\epsilon}$  was inside and
- 3)  $\epsilon$  was inside the control limits and  $\hat{\epsilon}$  was outside.

Figure 4.12: Counts of epsilon and its estimate for being inside and outside the control limits.

For  $\epsilon$  there will never be a false alarm generated. This is because the estimates for  $\epsilon$  are comparatively poor and are severely underestimating  $\epsilon$ .

The behaviour of our estimates appears to have some useful features when the process is under control i.e. the ARL-to-false-alarm is high. Now their behaviour will be compared when the process is not under control.

Suppose there has been a  $3\sigma_{z_1}$  shift in the mean of  $z_1$  i.e. the mean on  $z_1$  is now 3. That is equivalent to  $1.95\sigma_Y$  shift in  $Y$  when  $\rho = 0$ . 5000 values were generated for  $z_1$ ,  $z_2$  and  $\epsilon$  under this new distribution and were used to generate  $Y$ . Subsequently  $Y$  was used to generate  $\hat{z}_1$ ,  $\hat{z}_2$  and  $\hat{\epsilon}$ . An X-Chart for the first 50 values of  $Y$  when  $\rho = 0$  is in Figure 4.13

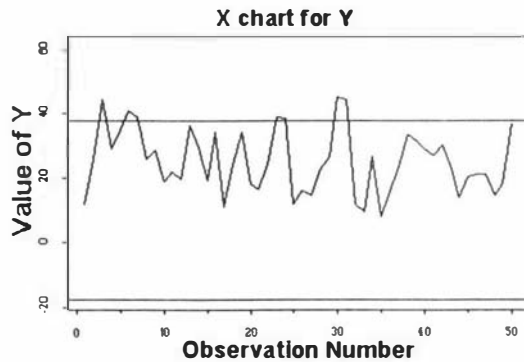
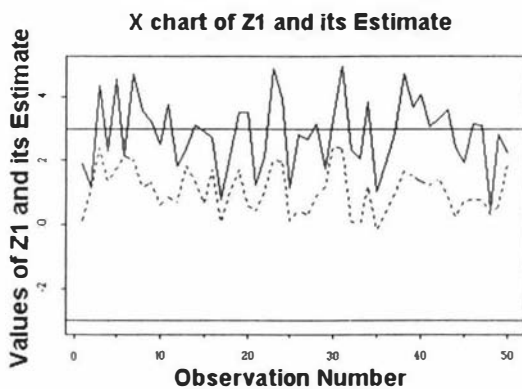


Figure 4.13: An X-Chart of the First 50 values of  $Y$  for  $\rho = 0$

The X-Chart of  $Y$  shows that there are about seven out of control points. For this shift of  $1.95\sigma_Y$  the ARL-to-detection was found to be 6.9. In [1] the ARL-to-detection for a  $2\sigma_Y$  shift was given as 6.

In Figure 4.14 is an X-chart of  $z_1$  and  $\hat{z}_1$ . The whole line represents  $z_1$  and the broken line  $\hat{z}_1$ . The former has clearly had a change in mean and the ARL-to-detection for this data is 2.01. This corresponds to the theoretical value of 2. This is lower than the ARL-to-detection for  $Y$  so being able to observe  $z_1$  would be invaluable.

In comparison the ARL-to-detection for  $\hat{z}_1$  is 238.10. This is clearly very poor.



The whole line represents  $z_1$ .

The jagged line represents  $\hat{z}_1$ .

Figure 4.14: An X-Chart of the First 50 values of  $\hat{z}_1$  for  $\rho = 0$

In Figure 4.15 there is an X-chart of  $z_2$  and  $\hat{z}_2$ . The whole line represents  $z_2$  and

the broken line  $\hat{z}_2$ .  $z_2$  has clearly had no change in mean. The ARL-to-false-alarm for  $z_2$  is 416.67 which is higher than the theoretical value of 370.

In comparison the ARL-to-false-alarm for  $\hat{z}_2$  is 50.51. This is clearly very poor.

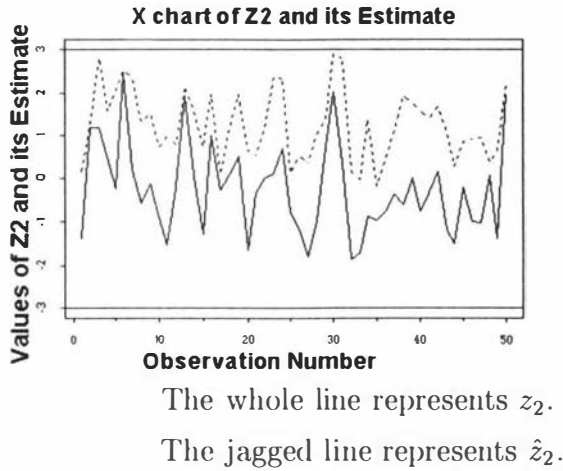


Figure 4.15: An X-Chart of the First 50 values of  $z_2$  and  $\hat{z}_2$  for  $\rho = 0$

In Figure 4.16 is an X-chart of  $\epsilon$  and  $\hat{\epsilon}$ . The whole line represents  $\epsilon$  and the broken line  $\hat{\epsilon}$ .  $\epsilon$  has clearly had no change in mean. The ARL-to-false-alarm is 384.61 which is close to the theoretical value of 370.

In comparison the ARL-to-false-alarm for  $\hat{\epsilon}$  is  $> 5000$  for this set of 5000 observations. This is because at this value of  $\rho$  the estimate  $\hat{\epsilon}$  is severely underestimating  $\epsilon$ .

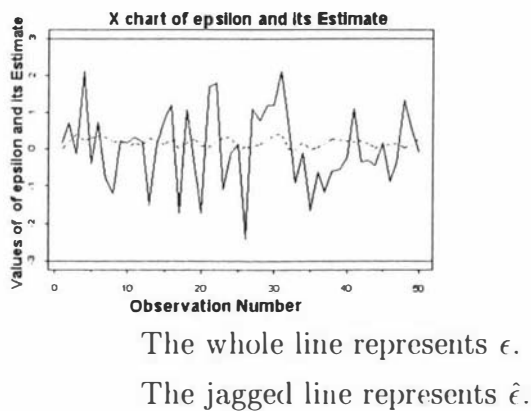


Figure 4.16: An X-Chart of the First 50 values of  $\epsilon$  and  $\hat{\epsilon}$  for  $\rho = 0$

X-Charts of  $\hat{z}_1$ ,  $\hat{z}_2$  and  $\epsilon$  are giving no more information about the behaviour of



the system than a X-Chart of  $Y$ . When the system was given a  $3\sigma_{z_1}$  shift in the mean of  $z_1$  then both  $\hat{z}_1$  and  $\hat{z}_2$  expressed some degree of that change rather than being solely reflected in the estimate of  $z_1$ . In fact the ARL-to-detection turned out to be lower for  $\hat{z}_2$  than ARL-to-detection for  $\hat{z}_1$ . This would mean a user could suspect that it was  $z_2$  that had a shift in mean rather than  $z_1$  i.e. an out of control condition is more likely to be observed in  $z_2$  than  $z_1$ . This is the opposite to what is wanted.

The reason that the ARL-to-false-alarm turned out to be lower for  $\hat{z}_2$  than ARL-to-detection for  $\hat{z}_1$  is in the way the estimates are calculated. Recalling that

$$\hat{z} = \frac{\Lambda B}{B^t \Lambda B + \sigma^2} (y - A)$$

Each element of  $\Lambda B$  is less than  $B^t \Lambda B + \sigma^2$ , the denominator in the estimate, so the estimates carve up  $Y - A$  in proportion to the fraction  $\frac{\Lambda B}{B^t \Lambda B + \sigma^2}$ . Now the denominator is a constant for each estimate so the part that changes from estimate to estimate is contained in the differences between the rows of  $\Lambda B$ . In the case in the example above  $\hat{z}_1$  always takes a 7% proportion of  $y - A$  and  $\hat{z}_2$  always takes an 8% proportion. This happens whether there is a change in mean of one of the variables or not. If the mean of one factor increases then  $Y - A$  increases and all the factors increase in order to still carve up  $y - A$  to the same percentage.

In this example  $\hat{z}_2$  is always going to be slightly larger than  $\hat{z}_1$ , because it carves up  $y - A$  with the bigger percentage, and therefore will be more likely to fall outside the control limits.

A similar situation occurs when a factor has had a change in variance rather than a change in mean. Using the data from the same example again but this time with an increase in the variance of  $\hat{z}_1$ . This would give an increase in the variance of  $Y - A$ . However there would still be the same proportions of  $y - A$  attributed to  $\hat{z}_1$  and  $\hat{z}_2$  i.e. 7% and 8% respectively. There is no ability in these estimates to be able to distinguish which factor out of  $\hat{z}_1$  and  $\hat{z}_2$  was associated with the variance change.

For these reasons an X-Charts based on  $\hat{z}$  and  $\hat{\epsilon}$  cannot be recommended.

It is very unfortunate that accurate estimates of  $z$  and  $\epsilon$  can't be found under process shifts in mean or variances. Their ARL-to-detection statistic is lower than for the same statistic for the variable  $Y$  while having the same value for ARL-to-false-alarm.

### 4.5.2 Comments

It is obviously very valuable to have some means of tracking the unobservable uncontrollable factors and detecting when they have had a change in mean or variance. The estimates presented here can, in some cases, provide very good estimates of the values of the unobservable uncontrollable factors. However comparing an X-chart of these estimates with an X-chart of the quality response showed that the former X-chart has larger ARL-to-false-alarm values. Also, if there is a change in the process, these estimates are unable to reflect which factors have changed and their resulting estimation ability deteriorates. Therefore X-charts of these estimates can't be recommended.

■

## Chapter 5

### Concluding Remarks

This thesis has demonstrated methods for utilising the information in the uncontrollable factors that has not been previously recognised in Parameter Design. These uncontrollable factors can be split into those that are not observable and those that are. Using the information available in the observable uncontrollable factors has allowed for choosing settings of the controllable factors that optimises the quality variable in some way. This could be minimising the variance while keeping the mean on target, minimising the Mean Square Error about the target or minimising an adjusted Mean Square Error.

The uncontrollable factors change their value over time. This means moving the controllable factors to a new setting to optimise the response. Methods were given for making these moves in order that a function of the quality response, taken over the complete movement, was minimised.

Finally the information that comes from observing the quality response, the controllable factors and the observable uncontrollable factors, allows for the estimation of the unobservable uncontrollable factors. Using these estimators as values on an X Chart were considered. However the X chart of these estimators was found to be less useful than an X chart of the response. The former had larger ARL-to-false-alarm values and failed to give information about which factors were out of control i.e. the charting of the estimators provides less information than charting the response.

■

# Appendix A

## Review Details

The review chapter of this thesis was originally written for my Masters thesis. It was used as a jumping off point for the work in the Ph.D. For this publication it has been updated and revised. The following table lists the changes made.

Section	Changes
1.1	Revision and additional material
1.2	Minor revisions and additional material
1.3	Very minor revisions
1.4	Revision and additional material
1.6	Additional material
1.7	New material
1.8	New material
1.9	Minor revisions

## Appendix B

### A Publication From This Thesis.

This paper, based on some of the work in chapter 2, was published in the Journal of Quality Technology, vol 28, 153-162.

# Observable Uncontrollable Factors in Parameter Design

MEGAN PLEDGER

*Massey University, Palmerston North, New Zealand*

Parameter design does not take into account the common occurrence that some of the uncontrollable factors are observable during production. The extra information any observable uncontrollable factors provides should enhance our choice of values for the controllable factors that both keeps the mean response on target and reduces the variance. This paper describes an approach to explicitly introduce uncontrollable factors into a designed experiment.

## Introduction

In his ideas for parameter design, Taguchi splits the factors into two sets. One set is referred to as the *controllable* factors and the other as the *uncontrollable* factors. Although several different methods of analysis have been used, the underlying idea is to choose levels for the controllable factors so that the uncontrollable factors have the least influence on the response. The mean response should also either hit some target, be minimized, or be maximized. In this paper the problem of hitting the target will be examined.

The different analysis methods fall into two groups depending on the way the uncontrollable factors are treated. Taguchi sees parameter design as a pre-production technique where the uncontrollable factors are designed into the experiment. His method uses inner and outer arrays and the S/N ratio. Box and Jones (1992) advocate the use of split-plots, while Vining and Myers (1990) and Myers, Khuri, and Vining (1992) use response surface methods. Box (1988), Engel (1992), and Nelder and Lee (1991) have given different methods of analysis where the uncontrollable factors need not explicitly enter into the design but are accounted for in the analysis by the replication of the design points of the controllable factors. A summary and discussion of these methods was given in Nair (1992).

Whatever the method of design, with the choice depending on the situation, the appropriate model is found in terms of the controllable factors. However,

---

Ms. Pledger is a doctoral student in the Statistics Department.

when the process is running some of the uncontrollable factors may be observable. This additional information should enhance our choice of the levels of the controllable factors. In this paper the uncontrollable factors are split into two sets, *observable* and *unobservable*. In the first set there may be factors like temperature and humidity, while in the second there may be component quality which may be unmeasurable due to expense. The aim is to find a relationship between the controllable factors and the observable uncontrollable factors while simultaneously minimizing the variance of the response and keeping the mean response on target. During the production process, whenever a particular value of an observable uncontrollable factor occurs, an appropriate value for the controllable factors can be generated that meets the stated objective.

In practice, the type of observable uncontrollable factors that would give the most benefits are the ones that change smoothly over time, like temperature and humidity, or ones that have a slower rate of influence than the production rate. Certainly, if an observable factor changes value in wild swings, it would not be sensible to make continuous invasive adjustments to the product or process. Rather, it would make sense to implement formal control over such a factor.

## Example

As an example consider the model below, which is an extension of the ideas in Myers, Khuri, and Vining (1992).

At time  $i$ , let  $y_i$  represent the response,  $x_i$  the controllable factors,  $t_i$  the observable uncontrollable

factors, and  $\mathbf{z}_i$  the unobservable uncontrollable factors. The aim here will be to minimize variance of  $y_i$ , while keeping the mean on target and finding a relationship between  $\mathbf{x}_i$  and  $\mathbf{t}_i$ .

Let the model for the quality response variable,  $y$ , be

$$y_i|\mathbf{x}_i, \mathbf{t}_i, \mathbf{z}_i = \beta_0 + \mathbf{x}'_i\beta_1 + \mathbf{t}'_i\beta_2 + \mathbf{z}'_i\beta_3 + \mathbf{x}'_i\Sigma_1\mathbf{t}_i + \mathbf{z}'_i\Sigma_2\mathbf{x}_i + \mathbf{z}'_i\Sigma_3\mathbf{t}_i + \epsilon_i \quad (1)$$

where the response is dependent on  $\mathbf{x}_i$ ,  $\mathbf{t}_i$ , and  $\mathbf{z}_i$ , and  $\beta_0$  is  $1 \times 1$ ,  $\beta_1$  is  $p \times 1$ ,  $\beta_2$  is  $q \times 1$ ,  $\beta_3$  is  $r \times 1$ ,  $\mathbf{x}_i$  is  $p \times 1$ ,  $\mathbf{t}_i$  is  $q \times 1$ ,  $\mathbf{z}_i$  is  $r \times 1$ ,  $\Sigma_1$  is  $p \times q$ ,  $\Sigma_2$  is  $r \times p$ ,  $\Sigma_3$  is  $r \times q$ ,  $\mathbf{t}_i \sim \text{Dist}(\mathbf{0}, \Lambda_1)$ ,  $\mathbf{z}_i \sim \text{Dist}(\mathbf{0}, \Lambda_2)$ , and  $\epsilon_i \sim \text{Dist}(0, \sigma_\epsilon^2)$ , where "Dist" represents some distribution. The last requirement is that  $\epsilon$ ,  $\mathbf{t}$ , and  $\mathbf{z}$  be independent.

Now

$$E_Y(Y_i|\mathbf{x}_i, \mathbf{t}_i) = E_Z(E_Y(Y_i|\mathbf{x}_i, \mathbf{t}_i, \mathbf{z}_i)) = \beta_0 + \mathbf{x}'_i\beta_1 + \mathbf{t}'_i\beta_2 + \mathbf{x}'_i\Sigma_1\mathbf{t}_i$$

and

$$\begin{aligned} \text{Var}_Y(Y_i|\mathbf{x}_i, \mathbf{t}_i) &= E_Z(\text{Var}_Y(Y_i|\mathbf{x}_i, \mathbf{t}_i, \mathbf{z}_i)) \\ &\quad + \text{Var}_Z(E_Y(Y_i|\mathbf{x}_i, \mathbf{t}_i, \mathbf{z}_i)) \\ &= \sigma_\epsilon^2 + \text{Var}_Z(\beta_0 + \mathbf{x}'_i\beta_1 + \mathbf{t}'_i\beta_2 \\ &\quad + \mathbf{x}'_i\Sigma_1\mathbf{t}_i + \mathbf{z}'_i(\beta_3 + \Sigma_2\mathbf{x}_i + \Sigma_3\mathbf{t}_i)) \\ &= \sigma_\epsilon^2 + (\beta_3 + \Sigma_2\mathbf{x}_i + \Sigma_3\mathbf{t}_i)' \Lambda_2 \\ &\quad \times (\beta_3 + \Sigma_2\mathbf{x}_i + \Sigma_3\mathbf{t}_i). \end{aligned}$$

We wish to minimize  $\text{Var}_Y(Y_i|\mathbf{x}_i, \mathbf{t}_i)$  subject to  $T = E_Y(Y_i|\mathbf{x}_i, \mathbf{t}_i)$ , where  $T$  is the target. Using Lagrangian minimization, we get

$$\mathbf{x}'_i = (2D'B)^{-1}(F + 2E'B)D - E \quad (2)$$

where, assuming that  $\mathbf{A}$  is invertible,

$$\begin{aligned} \mathbf{A} &= (\Sigma_2'\Lambda_2\Sigma_2)^{-1} \\ \mathbf{B} &= \frac{1}{2}(\beta_1 + \Sigma_1\mathbf{t}_i) \\ \mathbf{C} &= \Sigma_2'\Lambda_2(\beta_3 + \Sigma_3\mathbf{t}_i) \\ \mathbf{D} &= \mathbf{AB} \\ \mathbf{E} &= \mathbf{AC} \\ \mathbf{F} &= T - \beta_0 - \mathbf{t}'_i\beta_2. \end{aligned}$$

(See Appendix A for details.) So during production the values of the observable uncontrollable factors

$\mathbf{t}_i$  would be used with (2) obtain the values of the controllable factors,  $\mathbf{x}_i^*$ .

The usual method of parameter design is to treat  $\mathbf{t}_i$  and  $\mathbf{z}_i$  as one group and find a value for  $\mathbf{x}_i$  that keeps  $y_i$  on target with least variance. In this case

$$\begin{aligned} E_Y(Y_i|\mathbf{x}_i) &= E_T(E_Z(E_Y(Y_i|\mathbf{x}_i, \mathbf{t}_i, \mathbf{z}_i))) \\ &= \beta_0 + \mathbf{x}'_i\beta_1 \end{aligned}$$

and

$$\begin{aligned} \text{Var}_Y(Y_i|\mathbf{x}_i) &= E_T(E_Z(\text{Var}_Y(Y_i|\mathbf{x}_i, \mathbf{t}_i, \mathbf{z}_i))) \\ &\quad + \text{Var}_Z(E_Y(Y_i|\mathbf{x}_i, \mathbf{t}_i, \mathbf{z}_i)) \\ &\quad + \text{Var}_T(E_Y(Y_i|\mathbf{x}_i, \mathbf{t}_i)) \\ &= \sigma_\epsilon + (\beta_3 + \Sigma_2\mathbf{x}_i)' \Lambda_2(\beta_3 + \Sigma_2\mathbf{x}_i) \\ &\quad + \text{tr}(\Sigma_3'\Lambda_2\Sigma_3\Lambda_1) \\ &\quad + (\beta_2 + \Sigma_1'\mathbf{x}_i)' \Lambda_1(\beta_2 + \Sigma_1'\mathbf{x}_i). \quad (3) \end{aligned}$$

(See the results in Appendices B and C, respectively.)

Now we wish to minimize  $\text{Var}_Y(Y_i|\mathbf{x}_i)$  subject to  $T = E_Y(Y_i|\mathbf{x}_i)$ . Using Lagrangian minimization once more we get

$$\mathbf{x}_i^{**} = (\mathbf{K}'\beta_1)^{-1}(T - \beta_0 + \mathbf{J}'\beta_1)\mathbf{K} - \mathbf{J} \quad (4)$$

where, assuming  $\mathbf{G}$  is invertible,

$$\begin{aligned} \mathbf{G} &= (\Sigma_2'\Lambda_2\Sigma_2 + \Sigma_1\Lambda_1\Sigma_1')^{-1} \\ \mathbf{H} &= \Sigma_2'\beta_3 + \Sigma_1\beta_2 \\ \mathbf{K} &= \mathbf{G}\frac{\beta_1}{2} \\ \mathbf{J} &= \mathbf{GH}. \end{aligned}$$

(See Appendix C for details.)

Now for some values of  $\mathbf{t}_i$ ,  $\mathbf{x}_i^{**}$  may predict a value for  $y_i$  that is closer to the target than that predicted for  $\mathbf{x}_i^*$ . To compare the two predictions the statistic  $E((y_i - T)^2)$  will be considered.

For  $\mathbf{x}_i^*$ ,

$$E(y_i|\mathbf{x}_i^*, \mathbf{t}_i) = \beta_0 + \mathbf{x}_i^{*\prime}\beta_1 + \mathbf{t}'_i\beta_2 + \mathbf{x}_i^{*\prime}\Sigma_1\mathbf{t}_i$$

so

$$E(y_i - T)^2 = E(y - E(y|\mathbf{x}_i^*, \mathbf{t}_i))^2 = \text{Var}(y|\mathbf{x}_i^*, \mathbf{t}_i).$$

For  $\mathbf{x}_i^{**}$ ,

$$\begin{aligned} E(y_i|\mathbf{x}_i^{**}) &= \beta_0 + \mathbf{x}_i^{**\prime}\beta_1 \\ &= E(y_i|\mathbf{x}_i^{**}, \mathbf{t}_i) - (\mathbf{t}'_i\beta_2 + \mathbf{x}_i^{**\prime}\Sigma_1\mathbf{t}_i) \end{aligned}$$

OBSERVABLE UNCONTROLLABLE FACTORS IN PARAMETER DESIGN

155

so

$$\begin{aligned}
 E(y_i - T)^2 &= E(y_i - E(y_i | \mathbf{x}_i^{**}))^2 \\
 &= E(y_i - E(y_i | \mathbf{x}_i^{**}, t_i) + t_i' \beta_2 + \mathbf{x}_i^{**} \Sigma_1 t_i)^2 \\
 &= \text{Var}(y_i | \mathbf{x}_i^{**}, t_i) + (t_i' \beta_2 + \mathbf{x}_i^{**} \Sigma_1 t_i)^2.
 \end{aligned}$$

So far two methods have been specified (i.e., using  $\mathbf{x}_i^*$  or  $\mathbf{x}_i^{**}$  exclusively). A third approach is to use either method depending on the value of  $t_i$ . On observing a value of  $t_i$ , calculate  $E(y_i - T)^2$  for  $\mathbf{x}_i^*$  and  $\mathbf{x}_i^{**}$  and choose that corresponding to the least  $E(y_i - T)^2$  value.

In practice the model is specified, but the coefficients of the model are unknown and these are found by experimentation. These estimated coefficients are used to give estimates of  $y_i$ ,  $\mathbf{x}_i$ ,  $E(y_i)$ , and  $\text{Var}(y_i)$ , denoted by  $\hat{y}_i$ ,  $\hat{\mathbf{x}}_i$ ,  $\hat{E}(y_i)$ , and  $\widehat{\text{Var}}(y_i)$ , respectively.

Simulation

To illustrate these procedures let the underlying model be

$$\begin{aligned}
 y_i | \mathbf{x}_i, t_i, \mathbf{z}_i &= 100 + 5x_1 + 7x_2 - 4t_1 + 3t_2 - 5z_1 \\
 &\quad - 6z_2 - 4x_1t_1 + 6x_1z_1 - 7x_1z_2 - 3x_2t_2 \\
 &\quad - 7x_2z_2 + 8t_1z_1 - 8t_2z_2 + \epsilon \quad (5)
 \end{aligned}$$

where  $\epsilon$  is distributed as Normal ( $\mu = 0, \sigma^2 = 10$ ), and the target is  $T = 100$ . The first step in parameter design is to do an experiment with all the factors at fixed levels. This was done with a  $2^{6-1}$  factorial design, and the results are given in Table 1.

The model found was

$$\begin{aligned}
 \hat{y} &= 100.06 + 4.98x_1 + 6.45x_2 - 3.55t_1 + 3.74t_2 \\
 &\quad - 4.86z_1 - 6.85z_2 - 3.57x_1t_1 + 6.11x_1z_1 \\
 &\quad - 6.98x_1z_2 - 2.15x_2t_2 - 7.36x_2z_2 + 8.24t_1z_1 \\
 &\quad - 8.58t_2z_2.
 \end{aligned}$$

The residuals were used to estimate  $\sigma_\epsilon^2, \hat{\sigma}_\epsilon^2 = 11.56$ . Four simulations will be used to compare four different methods. In each simulation the same 100,000 standard normal random variates will be used for  $t_i, z_i$ , and  $\epsilon_i$ . The  $t_i$  and the  $z_i$  are standard normal, and  $\epsilon_i$  is distributed as  $N(\mu = 0, \sigma^2 = 10)$ .

For simulation 1, equation (2) was used to generate the  $\hat{\mathbf{x}}_i^*$  from the  $t_i$ . The equations for  $\hat{x}_1$  and  $\hat{x}_2$  in terms of  $t_i$  were generated in Maple (1994) and appear in Table 2. The  $\hat{\mathbf{x}}_i^*$  and  $t_i$  were used with the  $z_i$  to generate the  $\hat{y}_i^*$  using the regression model

TABLE 1. The Experimental Design and Responses for a  $2^{6-1}$  Factorial Design in  $x_1, x_2, t_1, t_2, z_1$ , and  $z_2$

Factors						Response
$x_1$	$x_2$	$t_1$	$t_2$	$z_1$	$z_2$	
-1	-1	-1	-1	-1	-1	91.5311
1	-1	-1	-1	-1	1	110.09547
-1	1	-1	-1	-1	1	120.90064
1	1	-1	-1	-1	-1	133.24822
-1	-1	1	-1	-1	1	99.8026
1	-1	1	-1	-1	-1	71.35239
-1	1	1	-1	-1	-1	101.25509
1	1	1	-1	-1	1	78.83396
-1	-1	-1	1	-1	1	106.93987
1	-1	-1	1	-1	-1	134.56421
-1	1	-1	1	-1	-1	138.35204
1	1	-1	1	-1	1	98.02461
-1	-1	1	1	-1	-1	96.73753
1	-1	1	1	-1	1	78.68531
-1	1	1	1	-1	1	92.89983
1	1	1	1	-1	-1	125.50157
-1	-1	-1	-1	1	1	79.88534
1	-1	-1	-1	1	-1	93.51969
-1	1	-1	-1	1	-1	80.66505
1	1	-1	-1	1	1	96.4498
-1	-1	1	-1	1	-1	58.77738
1	-1	1	-1	1	1	96.73602
-1	1	1	-1	1	1	101.59202
1	1	1	-1	1	-1	126.37272
-1	-1	-1	1	1	-1	75.88907
1	-1	-1	1	1	1	87.37854
-1	1	-1	1	1	1	66.28703
1	1	-1	1	1	-1	143.92734
-1	-1	1	1	1	1	93.3468
1	-1	1	1	1	-1	122.44309
-1	1	1	1	1	-1	116.39455
1	1	1	1	1	1	83.3959

in equation (5). The variance of  $\hat{y}_i^*$  was  $67.72 \pm 1.10$  and the mean  $100.40 \pm 0.03$ . The variance of  $\hat{y}_i^*$  can be broken into three components due to: (1)  $\epsilon$ , (2) the influence of the random variable  $z$  and its interaction with  $x$  and  $t$ , and (3) the use of the estimated coefficients. The size of the latter can be estimated by generating  $y$  with  $\epsilon$  and  $z_i$  equal to zero while keeping  $\hat{\mathbf{x}}_i^*$  and  $t_i$  as before. The resulting variance was 1.37. This variance would decrease with the increased accuracy of estimating the coefficients.

For the second simulation, equation (4) was used to obtain  $\hat{\mathbf{x}}_i^{**} = (0.0358, -0.0363)$  giving  $\widehat{\text{Var}}(y) = 249.40$  from equation (3). The underlying model is known in this case, so  $\text{Var}(y)$  can be calculated, and it is  $\text{Var}(y) = 223.67$ . Using  $\hat{\mathbf{x}}_i^{**}$  and the previous  $t_i$  and  $z_i$ , the  $\hat{y}_i^{**}$  were generated from the regression model in equation (5) to give a sampling variance of  $223.74 \pm 1.43$  with a mean of  $99.97 \pm 0.05$ .



TABLE 2. Maple (1994) Output giving the Relationship between the Controllable Factors and the Observable Uncontrollable Factors

$$\begin{aligned}
 &xstar := \\
 &[(- 6.920059986 - 4.847453307t_1 + 4.814515614t_1^2 \\
 &- .1496378740t_2 - 2.752304982t_2t_1 + 2.507256324t_2^2) \\
 &(.01532797490 + .04782552040t_1 - .02734032426t_2)/( \\
 &.4020299378 + .1094442124t_1 + .1707409638t_1^2 \\
 &- .3189447000t_2 - .1952143239t_2t_1 + .09853725602t_2^2) \\
 &+ .7962406109 - 1.348573357t_1] \\
 &[- (- 6.920059986 - 4.847453307t_1 + 4.814515614t_1^2 \\
 &- .1496378740t_2 - 2.752304982t_2t_1 + 2.507256324t_2^2) \\
 &(.07415264317 + .04538610643t_1 - .04581858850t_2)/( \\
 &.4020299378 + .1094442124t_1 + .1707409638t_1^2 \\
 &- .3189447000t_2 - .1952143239t_2t_1 + .09853725602t_2^2) \\
 &- 1.687182865 + 1.279787306t_1 - 1.165842752t_2]
 \end{aligned}$$

By accounting for the specific values of the  $t_i$  when choosing the  $\hat{x}_i$ , an approximately 70% decrease in variance was achieved. Histograms of the two data sets, shown in Figure 1, illustrate the reduction in the variance of the response  $\hat{y}_i$  from simulation 2 to simulation 1.

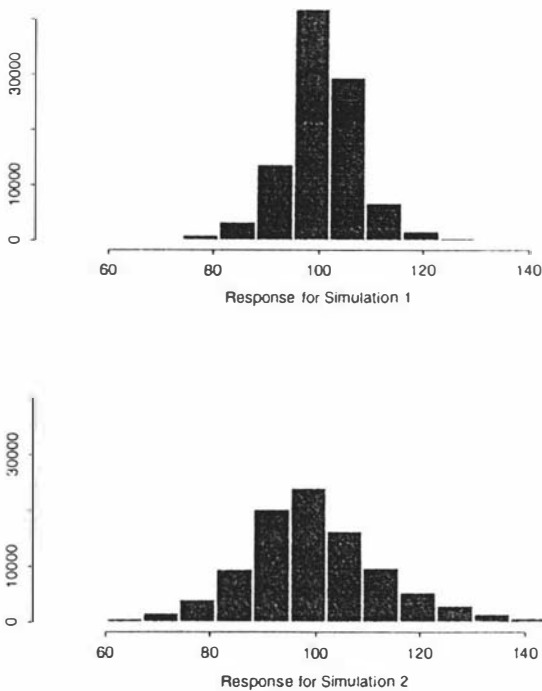


FIGURE 1. Histograms of the Responses Collected in Simulation 1 and Simulation 2, Showing a Reduction in Variance by Accounting for the Observable Uncontrollable Factors.

It is possible to generate a Box-type model (Box (1988)). In this method the uncontrollable factors do not enter into the experiment or the modeling process explicitly. During the experimental stage the design is replicated so that at each distinct design point there are multiple readings. The means and variances are collected at each point. They are both used as responses to be explained by the controllable factors. The  $2^{6-1}$  design was converted into a  $2^2$  design in  $x_1$  and  $x_2$  with 8 replications at each design point. The mean and variance at each design point were collected, and the model was

$$\begin{aligned}
 \hat{\mu} &= 100.06 + 4.98x_1 + 6.45x_2 - 0.76x_1x_2 \\
 \log(s) &= 3.04 + 0.10x_1 + 0.13x_2 - 0.07x_1x_2.
 \end{aligned}$$

Calculus shows the values of  $x_1$  and  $x_2$  that minimize  $\log(s)$  while keeping  $\hat{\mu}$  on a target of 100 are  $(-0.0026, -0.0067)$ . The model predicts a variance at that point of 433.87. Simulation 3, using these  $x_i$  with the previous  $t_i$  and  $z_i$ , gives a sampling variance of  $223.49 \pm 1.42$  and a mean of  $99.99 \pm 0.03$ . Again, accounting for  $t_i$  in calculating  $x_i$  has given an improvement over an accepted method.

The third approach is to calculate  $\hat{E}(y - T)^2$  for both  $\hat{x}_i^*$  and  $\hat{x}_i^{**}$  and to use the  $\hat{x}_i$  that minimizes  $\hat{E}(y - T)^2$ . Using Maple (1994), both expressions were derived in terms of  $t_1$  and  $t_2$ . A contour plot of  $\hat{E}(y|x_i^*, t_i - T)^2 - \hat{E}(y|x_i^{**}, t_i - T)^2 = 0$  is in Figure 2. The smaller regions denote where  $\hat{x}_i^*$  has lower  $\hat{E}(y - T)^2$  than  $\hat{x}_i^{**}$ .

A fourth simulation, using  $x_i^{**}$  when  $t_i$  falls in the enclosed regions and  $x_i^*$  otherwise, gives a mean

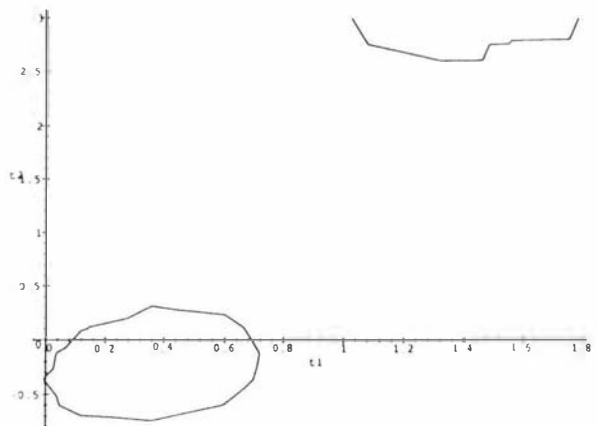


FIGURE 2. A Contour Plot of  $\hat{E}(y_i|x_i^*, t_i - T)^2 - \hat{E}(y_i|x_i^{**}, t_i - T)^2 = 0$ .

OBSERVABLE UNCONTROLLABLE FACTORS IN PARAMETER DESIGN

TABLE 3. Summary of the Simulation Results

Simulation	Mean	Standard Error of Mean	Variance	Standard Error of Variance
1	100.40	0.03	67.85	1.10
2	99.97	0.05	223.74	1.43
3	99.99	0.05	223.49	1.42
4	100.18	0.03	65.44	0.62

and variance for the response as  $100.18 \pm 0.03$  and  $65.44 \pm 0.62$ , respectively. This approach shows little improvement over using  $\hat{x}_i^*$  exclusively because only 8.9% of the  $t_i$  fall in the enclosed region. When the probability of the  $t_i$  falling in such regions is larger this approach may give greater improvements.

A summary of the four simulations appears in Table 3.

Complications

In the typical inner/outer array, only the main effects and the interactions between the controllable and uncontrollable factors can be estimated (i.e., everything in equation (1) except  $\Sigma_3$ ). However, especially when using response surface methods, it may be the case that an experiment is designed to produce a model containing interactions or quadratic terms or both. Adding terms like  $x_i' \Sigma x_i$  to the model allows no closed-form solution for  $x_i^*$ .

Other problems are that the  $x_i^*$  found may not be in the experimental region or that **A** or **G** or both are not invertible. These problems can be overcome with numerical solutions and appropriate upper and lower bounds on  $x_i$ .

A Second Example

Engel (1992) reports an experiment to improve an injection molding process by minimizing the shrink-

TABLE 4. The Factors in the Injection Molding Experiment

Controllable	Uncontrollable
A, Cycle Time	M, Percentage Regrind
B, Mold Temperature	N, Moisture Content
C, Cavity Thickness	O, Ambient Temperature
D, Holding Pressure	
E, Injection Speed	
F, Holding Time	
G, Gate Size	

age of the product. The experiment consists of the seven controllable factors and three uncontrollable factors listed in Table 4. The experiment design and the responses are in Table 5.

Of the noise factors, percent regrind appears to be unmeasurable while ambient temperature is measurable. Moisture content is assumed to be measurable, since it is a function of the ambient humidity and the amount of time the material is exposed to the air. All the factors appear quantitative.

Steinberg and Bursztyn (1994) re-analyzed this experiment and suggested that perhaps two of the responses had been swapped. These were 0.3 and 2.5 in the second row, first and second column. Assuming the responses were swapped, the experiment will be analyzed under the new method. For this example, the aim will be to set shrinkage to a low target rather than minimize it. The experiment can be written in the form give in Table 6.

A half-normal probability plot of the main effects and interactions appears in Figure 3. The effects and interactions not falling on the line were deemed to be significant, and a regression equation was derived with these. Note that the controllable factors B and F were found to be insignificant, so in production they can be set to our economic advantage.

TABLE 5. The Design and Responses for the Injection Molding Experiment

Cell	Controllable Factors							Percent Shrinkage for Noise Factors M, N, O			
	A	B	C	D	E	F	G	(1, 1, 1)	(1, 2, 2)	(2, 1, 2)	(2, 2, 1)
1	1	1	1	1	1	1	1	2.2	2.1	2.3	2.3
2	1	1	1	2	2	2	2	2.5	0.3	2.7	0.3
3	1	2	2	1	1	2	2	0.5	3.1	0.4	2.8
4	1	2	2	2	2	1	1	2.0	1.9	1.8	2.0
5	2	1	2	1	2	1	2	3.0	3.1	3.0	3.0
6	2	1	2	2	1	2	1	2.1	4.2	1.0	3.1
7	2	2	1	1	2	2	1	4.0	1.9	4.6	2.2
8	2	2	1	2	1	1	2	2.0	1.9	1.9	1.8

TABLE 6. The Rewritten Design for the Injection Molding Experiment

Factors											Response
A	B	C	D	E	F	G	M	N	O		
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	2.2
-1	-1	-1	1	1	1	1	1	-1	-1	-1	2.5
-1	1	1	-1	-1	1	1	1	-1	-1	-1	0.5
-1	1	1	1	1	-1	-1	-1	-1	-1	-1	2
1	-1	1	-1	1	-1	1	1	-1	-1	-1	3
1	-1	1	1	-1	1	1	-1	-1	-1	-1	2.1
1	1	-1	-1	1	1	1	-1	-1	-1	-1	4
1	1	-1	1	-1	-1	1	-1	-1	-1	-1	2
-1	-1	-1	-1	-1	-1	-1	-1	1	1	1	2.1
-1	-1	-1	1	1	1	1	1	-1	1	1	0.3
-1	1	1	-1	-1	1	1	1	-1	1	1	3.1
-1	1	1	1	1	-1	-1	-1	-1	1	1	1.9
1	-1	1	-1	1	-1	1	-1	1	1	1	3.1
1	-1	1	1	-1	1	-1	-1	1	1	1	4.2
1	1	-1	-1	1	1	-1	-1	1	1	1	1.9
1	1	-1	1	-1	-1	1	-1	1	1	1	1.9
-1	-1	-1	-1	-1	-1	-1	1	1	-1	1	2.3
-1	-1	-1	1	1	1	1	1	1	-1	1	2.7
-1	1	1	-1	-1	1	1	1	-1	1	1	0.4
-1	1	1	1	1	-1	-1	1	-1	1	1	1.8
1	-1	1	-1	1	-1	1	1	-1	1	1	3
1	-1	1	1	-1	1	-1	1	-1	1	1	1
1	1	-1	-1	1	1	-1	1	-1	1	1	4.6
1	1	-1	1	-1	-1	1	1	-1	1	1	1.9
-1	-1	-1	-1	-1	-1	-1	1	1	1	-1	2.3
-1	-1	-1	1	1	1	1	1	1	1	-1	0.3
-1	1	1	-1	-1	1	1	1	1	1	-1	2.8
-1	1	1	1	1	-1	-1	1	1	1	-1	2
1	-1	1	-1	1	-1	1	1	1	1	-1	3
1	-1	1	1	-1	1	-1	1	1	1	-1	3.1
1	1	-1	-1	1	1	-1	1	1	1	-1	2.2
1	1	-1	1	-1	-1	1	1	1	1	-1	1.8

The regression equation is

$$\hat{y} = 2.25 + 0.4250A + 0.0625C - 0.2813D + 0.1438E - 0.2313G - 0.05M + 0.1063ME - 0.125MC - 0.0938MD + 0.0000N + 0.5875NC - 0.5563NE. \tag{6}$$

Therefore

$$\hat{E}(y|A, C, D, E, G, N) = 2.25 + 0.4250A + 0.0625C - 0.2813D + 0.1437E - 0.2313G + 0.0000N + 0.5875NC - 0.55625NE.$$

In order to work out the variance, a couple of assumptions have to be made. Assume that the distributions of  $M$ , % regrind, and  $N$ , the ambient temperature,

are Normal and centered about zero. Further assume that 95% of the distribution lies between their levels (i.e., they have standard deviations of  $\frac{1}{2}$ ). Therefore

$$\widehat{\text{Var}}(y|A, C, D, E, G, N) = \text{Var}(M)(-0.05 - 0.125C + 0.1063E - 0.0938D)^2 + \hat{\sigma}_e^2.$$

In this case the target will be set at 1.6% shrinkage. A quadratic program was set up using Ampl (1993) with  $\widehat{\text{Var}}(y|A, C, D, E, G, N)$  as the objective that is to be minimized, subject to  $\hat{E}(y|A, C, D, E, G, N) = 1.6$ . Also, since the experiment was done in the unit hypercube, all controllable factors will be constrained to lie between  $\pm 1$ . During production, factor  $N$  can vary, and the best approach would be to solve the quadratic program for the particular value of  $N$  at that time. This case will be simulated later. Generally this will be expensive and approximations to the solution can be found. In the approximate method the idea will be to observe the solution to the quadratic program for different values of  $N$  and relate the solutions for the controllable factors to factor  $N$  by regression equations. Technically only values of  $N$  between  $\pm 1$  should be considered. However

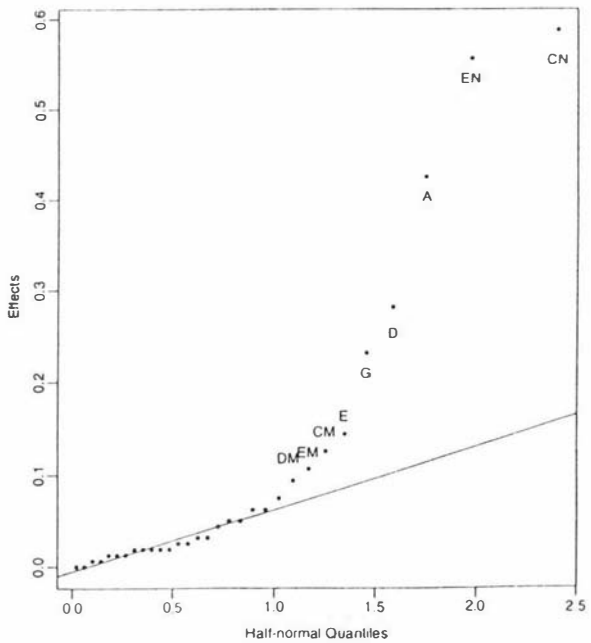


FIGURE 3. A Half-Normal Probability Plot of the Main Effects and the Interactions for the Injection Molding Experiment.

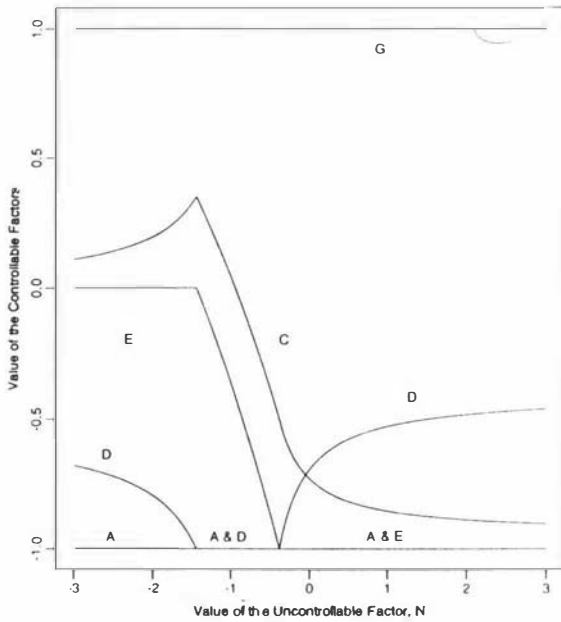


FIGURE 4. A Plot of the Observable Uncontrollable Factor N versus the Controllable Factors A, C, D, E, and G.

in practice values outside this range (and in this case 5 out of 100) will be seen, so  $N$  was observed over the range  $\pm 3$ . For each value of  $N$  a solution from the quadratic program was found for each of the controllable factors. These controllable factors were plotted against the uncontrollable factor  $N$  in Figure 4.

From this it can be seen that factor A should be always set at  $-1$  and factor G at  $1$ . The others change their values along smooth curves with breakpoints at  $-1.4425$  and  $-0.3825$ . Regression equations were fitted to these smooth curves in order that the values for factors C, D, and E could be predicted given a value of  $N$ . In this case we want these equations to be good predictors so that parsimony is not necessary. Terms were added to each equation until a term became insignificant or the residual squared error reached some threshold. In this case the threshold was  $1 \times 10^{-5}$ . This threshold could not be reached in the regression equation for  $-0.3825 \leq C \leq 3$ , and this equation does not predict as well as the others. The solution is in Table 7.

Since the underlying model is not known, we cannot tell how the solution would do. This could only be found by using the solution practically. However the two different methods can be compared to see if there is any difference between them. Two simulations were done with the same 100,000 random variates for each of  $N$  and  $M$ . In the first simulation, the quadratic program was solved for each value of  $N$  to give values for the controllable factors. These were used in equation (6) to generate the responses  $\hat{y}_1$ . The variance of  $\hat{y}_1$  was  $2.26 \times 10^{-12}$ . In the second simulation, the values of the controllable factors were chosen using the solution in Table 7, and these were used in equation (6) to generate the responses  $\hat{y}_2$ . The variance of  $\hat{y}_2$  was  $9.97 \times 10^{-11}$ . The two methods show little difference in the response variance. With any reasonable error variance these response

TABLE 7. Solution giving the Relationship between the Controllable Factors A, B, C, D, E, F, and G, and the Observable Uncontrollable Factor N that Minimizes the Variance and Keeps the Mean on Target.

Factor	Factor Value	Range of Factor N
A	$= -1$	$-3 < N < 3$
B	our choice	$-3 < N < 3$
C	$= \begin{cases} 19.31 + 59.20N + 83.18N^2 + \dots + 0.015N^8 \\ -0.93 - 1.25N - 0.34N^2 - 0.07N^3 - 0.01N^4 \\ -0.72 - 0.29N + 0.39N^2 + \dots + 0.0002N^{14} \end{cases}$	$-3 < N \leq -1.4425$ $-1.4425 < N \leq -0.3825$ $-0.3825 < N < 3$
D	$= \begin{cases} -26.28 - 78.90N - 110.85N^2 + \dots - 0.02N^8 \\ -1 \\ -0.70 + 0.39N - 0.52N^2 + \dots + 0.0003N^{13} \end{cases}$	$-2 < N \leq -1.4425$ $-1.4425 < N \leq -0.3825$ $-0.3825 < N \leq 3$
E	$= \begin{cases} 0 \\ -1.51 - 1.47N - 0.40N^2 - 0.09N^3 - 0.01N^4 \\ -1 \end{cases}$	$-2 < N \leq -1.4425$ $-1.4425 < N \leq -0.3825$ $-0.3825 < N \leq 3$
F	our choice	$-3 < N < 3$
G	$= 1$	$-3 < N < 3$

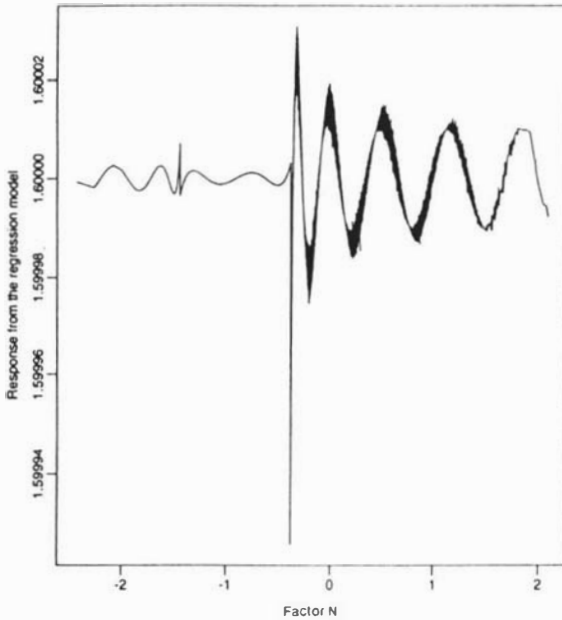


FIGURE 5. A Plot of the Response to the Regression Equation, using the Solutions in Table 7 for the Factors and the 100,000 Random Variates for  $M$  and  $N$ .

variances would be swamped and the two methods would be indistinguishable in outcome.

The responses for the second simulation were plotted against  $N$  in Figure 5. After  $N = -0.3825$ , the variance in the response is greater. This is due to the poorer predictive power of the regression model for factor  $C$  in this region.

**Conclusion**

Accounting for the observable uncontrollable factors in setting the values of the controllable factors can decrease the variance of the response while keeping the mean on target. The proposed approaches compare favorably with current methods of parameter design.

**Acknowledgments**

This paper was supported in part by the New Zealand Aluminium Smelters Ltd. The author wishes to thank the referees and the editor for their comments and suggestions which significantly improved this paper.

**Appendix A**

We wish to minimize  $\text{Var}_Y(Y_i|x_i, t_i)$  subject to  $T = E_Y(Y_i|x_i, t_i)$ , where  $T$  is the target. For this

we will use Lagrangian minimization with

$$L = x_i' \Sigma_2' \Lambda_2 \Sigma_2 x_i + 2\beta_3' \Lambda_2 \Sigma_2 x_i + 2x_i' \Sigma_2' \Lambda_2 \Sigma_3 t_i + \lambda(T - \beta_0 - x_i' \beta_1 - t_i \beta_2 - x_i' \Sigma_1 t_i)$$

$$\frac{\partial L}{\partial x_i} = 2\Sigma_2' \Lambda_2 \Sigma_2 x_i + 2\Sigma_2' \Lambda_2 \beta_3 + 2\Sigma_2' \Lambda_2 \Sigma_3 t_i - \lambda(\beta_1 + \Sigma_1 t_i)$$

therefore

$$x_i = (\Sigma_2' \Lambda_2 \Sigma_2)^{-1} \left( \frac{\lambda}{2} (\beta_1 + \Sigma_1 t_i) - \Sigma_2' \Lambda_2 (\beta_3 + \Sigma_3 t_i) \right)$$

Assuming that  $A$  is invertible, let

$$A = (\Sigma_2' \Lambda_2 \Sigma_2)^{-1}$$

$$B = \frac{1}{2} (\beta_1 + \Sigma_1 t_i)$$

$$C = \Sigma_2' \Lambda_2 (\beta_3 + \Sigma_3 t_i)$$

so  $x_i = A(\lambda B - C)$ . Let  $D = AB$ ,  $E = AC$ , and  $x_i = \lambda D - E$ . Now

$$\begin{aligned} \frac{\partial L}{\partial \lambda} &= T - \beta_0 - x_i' (\beta_1 + \Sigma_1 t_i) - t_i' \beta_2 \\ &= T - \beta_0 - 2(\lambda D' - E')B - t_i' \beta_2. \end{aligned}$$

Therefore  $\lambda = (2D'B)^{-1}(F + 2E'B)$  (where  $F = T - \beta_0 - t_i' \beta_2$ ) and  $x_i^* = (2D'B)^{-1}(F + 2E'B)D - E$ .

Additionally

$$\frac{\partial^2 L}{\partial x_i^2} = 2\Sigma_2' \Lambda_2 \Sigma_2.$$

As long as  $\Lambda_2$  is positive definite and  $\Sigma_2$  is of rank  $r$ , then  $x_i$  is the minimum critical point.

**Appendix B**

**Theorem**

$$\begin{aligned} \text{Var}(Y) &= E_X(E_Z(\text{Var}_Y(Y|X, Z))) \\ &\quad + E_X(\text{Var}_Z(E_Y(Y|X, Z))) \\ &\quad + \text{Var}_X(E_Y(Y|X)) \end{aligned}$$

**Proof**

It is known that

$$\text{Var}(Y) \leq E_X(\text{Var}_Y(Y|X)) + \text{Var}_X(E_Y(Y|X))$$

OBSERVABLE UNCONTROLLABLE FACTORS IN PARAMETER DESIGN

14

so

$$\text{Var}(Y|X) = E_X(\text{Var}_Y(Y|X, Z) + \text{Var}_Z(E_Y(Y|X, Z)))$$

giving the result

$$\text{Var}(Y) = E_X(E_Z(\text{Var}_Y(Y|X, Z) + \text{Var}_Z(E_Y(Y|X, Z))) + \text{Var}_X(E_Y(Y|X))).$$

Appendix C

In this case

$$E_Y = E_T(E_Z(E_Y(Y_i|x_i, t_i, z_i))) = \beta_0 + x_i'\beta_1$$

and

$$\begin{aligned} \text{Var}_Y(Y_i|x_i) &= E_T(E_Z(\text{Var}_Y(Y_i|x_i, t_i, z_i) + \text{Var}_Z(E_Y(Y_i|x_i, t_i, z_i)) + \text{Var}_T(E_Y(Y_i|x_i, t_i))) \\ &= \sigma_\epsilon + E_T((\beta_3 + \Sigma_2 x_i + \Sigma_3 t_i)'\Lambda_2 \times (\beta_3 + \Sigma_2 x_i + \Sigma_3 t_i) + \text{Var}_T(\beta_0 + x_i'\beta_1 + t_i'\beta_2 + x_i'\Sigma_1 t_i)). \end{aligned}$$

Firstly,

$$t_i'\Sigma_3'\Lambda_2\Sigma_3 t_i = \text{tr}(\Sigma_3'\Lambda_2\Sigma_3 t_i t_i')$$

so

$$E_T(t_i'\Sigma_3'\Lambda_2\Sigma_3 t_i) = \text{tr}(\Sigma_3'\Lambda_2\Sigma_3 \Lambda_1).$$

Now

$$\begin{aligned} E_T((\beta_3 + \Sigma_2 x_i + \Sigma_3 t_i)'\Lambda_2(\beta_3 + \Sigma_2 x_i + \Sigma_3 t_i)) &= (\beta_3 + \Sigma_2 x_i)'\Lambda_2(\beta_3 + \Sigma_2 x_i) + E_T(t_i'\Sigma_3'\Lambda_2\Sigma_3 t_i) + 2(\beta_3 + \Sigma_2 x_i)'\Sigma_3 t_i \\ &= (\beta_3 + \Sigma_2 x_i)'\Lambda_2(\beta_3 + \Sigma_2 x_i) + E_T(t_i'\Sigma_3'\Lambda_2\Sigma_3 t_i) \\ &= (\beta_3 + \Sigma_2 x_i)'\Lambda_2(\beta_3 + \Sigma_2 x_i) + \text{tr}(\Sigma_3'\Lambda_2\Sigma_3 \Lambda_1). \end{aligned}$$

And finally,

$$\begin{aligned} \text{Var}(\beta_0 + x_i'\beta_1 + t_i'\beta_2 + x_i'\Sigma_1 t_i) &= \text{Var}(\beta_0 + x_i'\beta_1 + t_i'(\beta_2 + \Sigma_1' x_i)) \\ &= (\beta_2 + \Sigma_1' x_i)'\Lambda_1(\beta_2 + \Sigma_1' x_i). \end{aligned}$$

Substituting,

$$\begin{aligned} \text{Var}_Y(Y_i|x_i) &= \sigma_\epsilon + E_T((\beta_3 + \Sigma_2 x_i + \Sigma_3 t_i)'\Lambda_2 \times (\beta_3 + \Sigma_2 x_i + \Sigma_3 t_i) + \text{Var}(\beta_0 + x_i'\beta_1 + t_i'\beta_2 + x_i'\Sigma_1 t_i)) \\ &= \sigma_\epsilon + (\beta_3 + \Sigma_2 x_i)'\Lambda_2(\beta_3 + \Sigma_2 x_i) + \text{tr}(\Sigma_3'\Lambda_2\Sigma_3 \Lambda_1) + (\beta_2 + \Sigma_1' x_i)'\Lambda_1(\beta_2 + \Sigma_1' x_i). \end{aligned}$$

We wish to minimize  $\text{Var}_Y(Y_i|x_i)$ , subject to  $T = E_Y(Y_i|x_i)$ . Using Lagrangian minimization once more

$$L = x_i'\Sigma_2'\Lambda_2\Sigma_2 x_i + x_i'\Sigma_1\Lambda_1\Sigma_1' x_i + 2\beta_3'\Sigma_2 x_i + 2\beta_2'\Sigma_1' x_i + \lambda(T - \beta_0 - x_i'\beta_1)$$

$$\begin{aligned} \frac{\partial L}{\partial x_i} &= 2\Sigma_2'\Lambda_2\Sigma_2 x_i + 2\Sigma_1\Lambda_1\Sigma_1' x_i + 2\Sigma_2'\beta_3 \\ &+ 2\Sigma_1\beta_2 - \lambda\beta_1 \end{aligned}$$

and

$$x_i = (\Sigma_2'\Lambda_2\Sigma_2 + \Sigma_1\Lambda_1\Sigma_1')^{-1}(\frac{\lambda}{2}\beta_1 - \Sigma_2'\beta_3 - \Sigma_1\beta_2).$$

Providing **G** is invertible, let

$$\begin{aligned} \mathbf{G} &= (\Sigma_2'\Lambda_2\Sigma_2 + \Sigma_1\Lambda_1\Sigma_1')^{-1} \\ \mathbf{H} &= \Sigma_2'\beta_3 + \Sigma_1\beta_2 \end{aligned}$$

so  $x_i = \mathbf{G}(\frac{\lambda}{2}\beta_1 - \mathbf{H})$ . Let

$$\begin{aligned} \mathbf{K} &= \mathbf{G}\frac{\beta_1}{2} \\ \mathbf{J} &= \mathbf{G}\mathbf{H} \end{aligned}$$

and  $x_i = \lambda\mathbf{K} - \mathbf{J}$ . Then

$$\begin{aligned} \frac{\partial L}{\partial \lambda} &= T - \beta_0 - x_i'\beta_1 \\ &= T - \beta_0 - (\lambda\mathbf{K}' - \mathbf{J}')\beta_1 \end{aligned}$$

so

$$\lambda = (\mathbf{K}'\beta_1)^{-1}(T - \beta_0 + \mathbf{J}'\beta_1)$$

and

$$x_i^* = (\mathbf{K}'\beta_1)^{-1}(T - \beta_0 + \mathbf{J}'\beta_1)\mathbf{K} - \mathbf{J}.$$

Additionally

$$\frac{\partial^2 L}{\partial x_i^2} = 2\Sigma_2'\Lambda_2\Sigma_2 + 2\Sigma_1\Lambda_1\Sigma_1'.$$

162

MEGAN PLEDGER

As long as  $\Lambda_2$  and  $\Lambda_1$  are both positive definite and  $\Sigma_2$  and  $\Sigma_1$  are of rank  $r$  and  $q$  respectively, then  $x_i^*$  is the minimum critical point.

### References

- AMPL, STUDENT VERSION (1993). *AMPL - A Modelling Language for Mathematical Programming*. The Scientific Press, San Francisco, CA.
- BOX, G. E. P. (1988). "Signal-to-Noise Ratios, Performance Criteria, and Transformations" (with discussion). *Technometrics* 30, pp. 1-40.
- BOX, G. E. P. and JONES, S. (1992). "Split-plot Designs For Robust Product Estimation". *Journal of Applied Statistics* 19, pp. 3-26.
- ENGEL, J. (1992). "Modeling Variation in Industrial Experiments". *Journal of Applied Statistics* 41, pp. 579-593.
- MAPLE V, RELEASE 2 (1994). *First Leaves Language Reference Manual*. Waterloo Maple Software, Waterloo, ON, Canada.
- MYERS, R. H.; KHURI, A. I.; and VINING, G. (1992). "Response Surface Alternatives to the Taguchi Robust Parameter Design". *The American Statistician* 46, pp. 131-139.
- NAIR, V. N. (EDITOR) (1992). "Taguchi's Parameter Design: A Panel Discussion". *Technometrics* 34, pp. 128-161.
- NELDER, J. A. and LEE, Y. (1991). "Generalized Linear Models For The Analysis Of Taguchi-Type Experiments". *Applied Stochastic Models and Data Analysis* 7, pp. 107-120.
- STEINBERG, D. M. and BURSZTYN, D. (1994). "Dispersion Effects in Robust-Design Experiments with Noise Factors". *Journal of Quality Technology* 26, pp. 12-20.
- VINING, G. G. and MYERS, R. H. (1990). "Combining Taguchi and Response Surface Philosophies: A Dual Response Approach". *Journal of Quality Technology* 22, pp. 38-45.

Key Words: *Design of Experiments, Optimization, Parameter Design, Taguchi Methods.*

## Appendix C

# Source Files For The Quadratic Program

The files in this section are used to solve a quadratic program using the package Student Ampl. The Model and Data files described below were originally set up by Dr Julie Faulkner, formerly of Massey University, who introduced me to the Student Ampl package. The problem consists of minimising a variance function subject to the mean being on target and with bounded parameters.

The files are

- 1) A \*.mod file - a model file that sets up the structure of the problem. The objective is a variance function which is to be minimised subject to the constraint that the mean is on target. The number of parameters and the values of the objective and the constraints are not specified in this file. This file remains unchanged from problem to problem.
- 2) A \*.dat file - a data file which sets out the number of parameters and the values in the objective and constraints. This specific data file is for the data in the Engel problem of Chapter 2.
- 3) A \*.com file - a command file that instructs ampl to solve each quadratic program for each inputted value of the observable uncontrollable factors and to output the results. This specific file is for the Engel problem of Chapter 2.
- 4) A \*.gen file - a generator file of Splus commands to write the \*.com file above.
- 5) A \*.ans file - an answer file of output for the Engel example of Chapter 2.



## Model File

A model file setting up the structure of the generic problem. The file is called `megan.mod`

```
# Model file called Megan.mod

param m > 0;           # number of unknown variables
param n > 0;           # number of known parameters

param ub;             # upper bound on variables and parameters
param lb <= ub;       # lower bound on variables and parameters

set index := 1..m+n;   # index-set for coefficients
set vindex := 1..m;    # index-set for variables
set pindex := m+1..m+n; # index-set for known parameters

param const;          # the constant term in the constraint
param alpha{index};   # coefficients of linear terms in constraint
param beta{index,index}; # coeffs of quadratic terms in constraint
param gamma{index};   # coeffs of linear terms in objective
param delta{index,index}; # coeffs of quadratic terms in objective

var xv{vindex} >= lb <= ub ; # the variables

param xp{pindex} >= -3.1 <= 3.1; # values of the known parameters

# the objective function
minimize cost: sum{i in index} gamma[i] *
                (if i <= m then xv[i] else xp[i])
                + sum{i in index, j in index} delta[i,j] *
                (if i <= m then xv[i] else xp[i]) *
                (if j <= m then xv[j] else xp[j]);

# the constraint
subject to quad: sum{i in index} alpha[i] *
```

```

                (if i <= m then xv[i] else xp[i])
+  sum{i in index, j in index} beta[i,j] *
                (if i <= m then xv[i] else xp[i]) *
                (if j <= m then xv[j] else xp[j])
= const;

```

## Data File

The data file gives the number of parameters and the values in the objective and the constraint.

# Data file called engelchange.dat

```

param m :=5;           # number of unknown variables
param n := 1;          # number of known parameters
param const := -.65;   # the constant term in the constraint
param lb := -1.;      # the lower bound
param ub := 1.;       # the upper bound

```

```

param alpha :=         # coeffs of linear terms in the constraint
  1    0.4250
  2    0.0625
  3   -0.28125
  4    0.14375
  5   -0.23125
  6    0;

```

```

param beta :          # coeffs of quadratic terms in the constraint
      1    2    3    4    5    6:=
  1    0    0    0    0    0    0
  2    0    0    0    0    0    0.29375
  3    0    0    0    0    0    0
  4    0    0    0    0    0   -0.278125
  5    0    0    0    0    0    0
  6    0    0.29375 0   -0.278125 0    0;

```

```

param gamma :=          # coeffs of linear terms in the objective
1  0
2  0.003125
3  0.00234375
4  -0.00265625
5  0
6  0;

```

```

param delta :          # coeffs of quadratic terms in the objective
      1      2          3          4          5      6:=
1  0      0          0          0          0      0
2  0      0.00390625  0.0029296875  -0.0033203125  0      0
3  0      0.0029296875  0.002197265625  -0.002490234375  0      0
4  0      -0.0033203125  -0.002490234375  0.002822265625  0      0
5  0      0          0          0          0      0
6  0      0          0          0          0      0;

```

## Command File

This file is run from within `ampl` and instructs `ampl` to solve the quadratic program for each  $xp[6]$ , the one observable, uncontrollable factor, from -3 to 3 in step sizes of 0.025. This list is a very small snippet of the actual file.

```

reset;
model megan.mod;
data engelchange.dat;
let xp[6]:= -3;
solve;
display xv >engel.ans;

let xp[6]:= -2.975;
solve;
display xv >engel.ans;

```

```
let xp[6] := -2.95;
solve;
display xv >engel.ans;
```

```
let xp[6] := -2.925;
solve;
display xv >engel.ans;
```

etc...

## Gen File

This file generates the file `engel.com` listed above. The above file is too long to be generated efficiently by hand.

```
function()
{
  cat("reset;", fill = T, file = "engel.com")
  cat("model megan.mod;", fill = T, file = "engel.com", append = T)
  cat("data engelchange.dat;", fill = T, file = "engel.com",
      append = T)
  for(i in seq(-3, 3, by = 0.025)) {
    cat("let xp[6] := ", i, ";;", fill = T, file = "engel.com",
        append = T)
    cat("solve;", fill = T, file = "engel.com", append = T)
    cat("display xv >engel.ans;", fill = T, file = "engel.com",
        append = T)
    cat(" ", fill = T, file = "engel.com", append = T)
  }
}
```

## Output File

This file is a very small snippet of the output file generated by `engel.com`. After it's stripped of extraneous characters it can be read into `Splus` or some other

mathematical package for manipulation.

xv gives the values of the controllable factors that should be used for each value of xp, the one observable, uncontrollable factor.

```
xv [*] :=
```

```
1 -1
2 -0.40000134951903465
3 0
4 0
5 0.0317516638867985
;
```

```
xv [*] :=
```

```
1 -1
2 -0.3999689699064476
3 0
4 0
5 0.0317516638867985
;
```

```
xv [*] :=
```

```
1 -0.9999350776809466
2 -0.39999999467610153
3 0
4 0
5 0.0317516638867985
;
```

```
etc
```



## Appendix D

# Maple And Splus Commands

This section gives the Maple and Splus code needed to find the mixture path as an approximation to the optimal path. The code is for a specific problem but it is easily amended for different data and for a larger number of variables.

### Maple Commands

These are the Maple commands for generating the mixture path for

$$L(x_1, x_2) = x_1^2 + x_2^2 - x_1x_2 \text{ with } x_0^t = (1, 2) \text{ and } x^{*t} = (0, 0).$$

```
> with(linalg):
```

```
Warning: new definition for norm
```

```
Warning: new definition for trace
```

```
> x0:=matrix(2,1,[1,2]);
```

```

           [ 1 ]
x0 := [   ]
           [ 2 ]
```

```
> M:=matrix(2,2,[2,-1,-1,2]);
```

```

           [ 2  -1 ]
M := [         ]
           [ -1  2 ]
```

```
> N:=matrix(2,1,[-0,0]);
```

$$N := \begin{bmatrix} 0 \\ \\ 0 \end{bmatrix}$$

```
> II:=matrix(2,2,[1,0,0,1]);
```

$$II := \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

```
> xstar:=matrix(2,1,[0,0]);
```

$$xstar := \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

```
> Ma:=evalm(-a*M-(1-a)*II);
```

$$Ma := \begin{bmatrix} -a-1 & a \\ a & -a-1 \end{bmatrix}$$

```
> Na:=evalm(-a*N+(1-a)*xstar);
```

$$Na := \begin{bmatrix} 0 \\ \\ 0 \end{bmatrix}$$

```
> P1:=evalm(exponential(Ma*q)&*x0);
```

$$P1 := \begin{bmatrix} 3/2 \exp(-q) - 1/2 \exp(-2qa-q) \\ \\ \end{bmatrix}$$

$$\left[ \frac{1}{2} \exp(-2qa - q) + \frac{3}{2} \exp(-q) \right]$$

```
> P2:=evalm(exponential(Ma*(q-s))*Na);
```

```
bytes used=1000124, alloc=720764, time=2.12
```

```
[ 0 ]
```

```
P2 := [ ]
```

```
[ 0 ]
```

```
> P21:=int(P2[1,1],s=0..q);
```

```
P21 := 0
```

```
> P21:=numer(P21)/denom(P21);
```

```
P21 := 0
```

```
> P22:=int(P2[2,1],s=0..q);
```

```
P22 := 0
```

```
> P22:=numer(P22)/denom(P22);
```

```
P22 := 0
```

```
> #first path
```

```
> x:=evalm(P1[1,1]+P21);
```

$$x := \frac{3}{2} \exp(-q) - \frac{1}{2} \exp(-2qa - q)$$

```
> #second path
```

```
> y:=evalm(P1[2,1]+P22);
```

$$y := \frac{1}{2} \exp(-2qa - q) + \frac{3}{2} \exp(-q)$$



```
> dotx:=diff(x,q);
```

$$\text{dotx} := -3/2 \exp(-q) - 1/2 (-2a - 1) \exp(-2qa - q)$$

```
> doty:=diff(y,q);
```

$$\text{doty} := 1/2 (-2a - 1) \exp(-2qa - q) - 3/2 \exp(-q)$$

## Splus Functions for Finding $\alpha$

There are two functions

**theint.f**: which represents the integrand of the line integral. It takes inputs of  $q$ , the variable of integration, and  $a$ , the trial version of  $\alpha$ .

**tominint.f**: integrates the integral whose integrand is *theint.f*. It takes inputs  $a$  as the trial version of  $\alpha$ .

To find  $\alpha$  the command

```
nmin(tominint.f,0.7,print.level=1)
```

is called. This Splus function *nmin* minimises the function *tominint.f* with respect to  $a$ . The value 0.7 is the starting value for  $a$ .

### The Function:tominint.f

```
>tominint.f<-
```

```
function(a)
```

```
{
```

```
  integrate(theint.f, lower = 0, upper = Inf, a = a)$integral
```

```
}
```

### The Function:theint.f

```
> theint.f
```

```
function(q, a)
```

```
{
```

```
  x <- 3/2 * exp(-q) - 1/2 * exp(-q - 2 * q * a)
```

```
  y <- 1/2 * exp(-q - 2 * q * a) + 3/2 * exp(-q)
```

```
  dotx <- -3/2 * exp(-q) - 1/2 * (-1 - 2 * a)
```

```

      * exp( - q - 2 * q * a)
doty <- 1/2 * (-1 - 2 * a) * exp( - q - 2 * q * a)
      - 3/2 * exp( - q)
(x^2 + y^2 - x * y) * (dotx^2 + doty^2)^(1/2)
}

```

## Output

The output of the command `nlmin(tominint.f,0.7,print.level=1)` is given below.

```

nlmin(tominint.f,0.7,print.level=1)
  it   nf     f      reldf      preldf      reldx
  0     1  2.159e0
  1     2  2.157e0   1.068e-3   5.972e-4   3.764e-2
  2     3  2.154e0   1.208e-3   1.696e-3   1.649e-1
  3     4  2.154e0   1.576e-4   2.033e-4   5.440e-2
  4     5  2.154e0   6.831e-6   6.393e-6   8.444e-3
  5     6  2.154e0   2.797e-8   2.846e-8   5.878e-4
  6     7  2.154e0   6.981e-12  6.974e-12  9.135e-6
***** relative function convergence *****
function      2.153641e0   reldx          9.1346e-6
function evaluations:  7   gradient evaluations:  9
preldf        6.9739e-12  npreldf       6.9739e-12
$x:
[1] 0.509677143334584576

$converged:
[1] T

$conv.type:
[1] "relative function convergence"

```

After 6 steps `nlmin` finds a convergent solution for  $\alpha$  and returns  $\alpha = 0.5097$  (4 dpl).

To calculate the line integral corresponding to the straight line path just requires calling `tominint.f` with  $a = 0$ .

```
> tominint.f(0)
```

```
[1] 2.23606797735315776
```

To calculate the line integral corresponding to the greedy path just requires calling *tominint.f* with  $a = 1$ .

```
> tominint.f(1)
```

```
[1] 2.18111585905441568
```

To calculate the line integral corresponding to the mixture path integral just requires calling *tominint.f* with  $a = 0.509677143334584576$ .

```
tominint.f(0.509677143334584576)
```

```
[1] 2.1536410551100912
```

For this case, the order of preference for the choice of path is (1) the mixture path, (2) the greedy path and (3) the straight line path.

■

# Appendix E

## Approximation Data

This data was used to create an approximate solution for the optimal path from  $(x_0, y_0)$  to the minimum on the surface of  $L = a_1x^2 + a_3y^2$ . The data is

column 1: is the initial  $x$  position,  $x_0$

column 2: is the initial  $y$  position,  $y_0$

column 3: is the value for  $a_1$

column 4: is the value for  $a_3$

The optimal path for each  $x_0, y_0, a_1$  and  $a_3$  was found in terms of  $x$  and  $y$  as output from the function *optfinddiff.f* discussed in appendix F.

column 5: is the value of the constant when  $\log(y)$  is fitted against  $\log(x)$  for the path found using the specified values of  $x_0, y_0, a_1$  and  $a_3$ .

column 6: is the value of linear terms when  $\log(y)$  is fitted against  $\log(x)$  for the path found using the specified values of  $x_0, y_0, a_1$  and  $a_3$ .

Columns	1	2	3	4	5	6	Columns	1	2	3	4	5	6
1	1	1	1	1	-1.387102e-016	1	2	1	4	1	1	1.386294	1
3	1	7	1	1	1.945910	1	4	1	10	1	1	2.302585	1
5	4	1	1	1	-1.386294	1	6	4	4	1	1	-3.467754e-016	1
7	4	7	1	1	5.596158e-001	1	8	4	10	1	1	9.162907e-001	1
9	7	1	1	1	-1.945910	1	10	7	4	1	1	-5.596158e-001	1
11	7	7	1	1	2.774203e-016	1	12	7	10	1	1	3.566749e-001	1
13	10	1	1	1	-2.302585	1	14	10	4	1	1	-9.162907e-001	1
15	10	7	1	1	-3.566749e-001	1	16	10	10	1	1	-2.774203e-016	1
17	1	1	1	1	-1.387102e-016	1	18	1	4	1	1	1.386294	1
19	1	7	1	1	1.945910	1	20	1	10	1	1	2.302585	1
21	4	1	1	1	-1.386294	1	22	4	4	1	1	-3.467754e-016	1
23	4	7	1	1	5.596158e-001	1	24	4	10	1	1	9.162907e-001	1
25	7	1	1	1	-1.945910	1	26	7	4	1	1	-5.596158e-001	1
27	7	7	1	1	2.774203e-016	1	28	7	10	1	1	3.566749e-001	1
29	10	1	1	1	-2.302585	1	30	10	4	1	1	-9.162907e-001	1
31	10	7	1	1	-3.566749e-001	1	32	10	10	1	1	-2.774203e-016	1
33	1	1	1	4	-5.009483e-002	2.399414	34	1	4	1	4	1.295189	2.486314
35	1	7	1	4	1.856405	2.529466	36	1	10	1	4	2.218388	2.556390
37	4	1	1	4	-3.295319	2.370107	38	4	4	1	4	-1.990094	2.399414
39	4	7	1	4	-1.493846	2.428429	40	4	10	1	4	-1.180343	2.451794

Columns	1	2	3	4	5	6	Columns	1	2	3	4	5	6
41	7	1	1	4	-4.613235	2.367877	42	7	4	1	4	-3.275105	2.381254
43	7	7	1	4	-2.773229	2.399414	44	7	10	1	4	-2.465525	2.416707
45	10	1	1	4	-5.455371	2.367298	46	10	4	1	4	-4.099219	2.374703
47	10	7	1	4	-3.584576	2.386614	48	10	10	1	4	-3.272364	2.399414
49	1	1	2	4	-4.962620e-003	1.576490	50	1	4	2	4	1.381293	1.604032
51	1	7	2	4	1.943357	1.611772	52	1	10	2	4	2.301271	1.614863
53	4	1	2	4	-2.169246	1.565258	54	4	4	2	4	-8.041474e-001	1.576490
55	4	7	2	4	-2.614558e-001	1.587326	56	4	10	2	4	8.485774e-002	1.594962
57	7	1	2	4	-3.043077	1.564468	58	7	4	2	4	-1.669582	1.569412
59	7	7	2	4	-1.126760	1.576490	60	7	10	2	4	-7.844333e-001	1.583092
61	10	1	2	4	-3.600466	1.564266	62	10	4	2	4	-2.222180	1.566935
63	10	7	2	4	-1.675567	1.571486	64	10	10	2	4	-1.332380	1.576490
65	1	1	3	4	4.909546e-004	1.212109	66	1	4	3	4	1.386899	1.215756
67	1	7	3	4	1.946585	1.216205	68	1	10	3	4	2.303280	1.216328
69	4	1	3	4	-1.676104	1.209707	70	4	4	3	4	-2.935548e-001	1.212109
71	4	7	3	4	2.634529e-001	1.213958	72	4	10	3	4	6.188387e-001	1.214933
73	7	1	3	4	-2.352674	1.209526	74	7	4	3	4	-9.688057e-001	1.210638
75	7	7	3	4	-4.122544e-001	1.212109	76	7	10	3	4	-5.795029e-002	1.213300
77	10	1	3	4	-2.783962	1.209480	78	10	4	3	4	-1.399224	1.210087
79	10	7	3	4	-8.421058e-001	1.211087	80	10	10	3	4	-4.879085e-001	1.212109
81	1	1	4	4	-1.387102e-016	1	82	1	4	4	4	1.386294	1
83	1	7	4	4	1.945910	1	84	1	10	4	4	2.302585	1
85	4	1	4	4	-1.386294	1	86	4	4	4	4	-3.467754e-016	1
87	4	7	4	4	5.596158e-001	1	88	4	10	4	4	9.162907e-001	1
89	7	1	4	4	-1.945910	1	90	7	4	4	4	-5.596158e-001	1
91	7	7	4	4	2.774203e-016	1	92	7	10	4	4	3.566749e-001	1
93	10	1	4	4	-2.302585	1	94	10	4	4	4	-9.162907e-001	1
95	10	7	4	4	-3.566749e-001	1	96	10	10	4	4	-2.774203e-016	1
97	1	1	1	7	-1.367470e-001	3.289976	98	1	4	1	7	1.116551	3.408096
99	1	7	1	7	1.646999	3.473312	100	1	10	1	7	1.993694	3.517960
101	4	1	1	7	-4.539487	3.247863	102	4	4	1	7	-3.311327	3.289976
103	4	7	1	7	-2.865656	3.328609	104	4	10	1	7	-2.587535	3.359947
105	7	1	1	7	-6.338714	3.244182	106	7	4	1	7	-5.049362	3.264793
107	7	7	1	7	-4.592834	3.289976	108	7	10	1	7	-4.319807	3.313052
109	10	1	1	7	-7.4905154172	3.243201	110	10	4	1	7	-6.1663497050	3.255081
111	10	7	1	7	-5.6896922864	3.272417	112	10	10	1	7	-5.4096108617	3.289976
113	1	1	3	7	-0.0103862314	1.736302	114	1	4	3	7	1.3730415032	1.777213
115	1	7	3	7	1.9369035545	1.791531	116	1	10	3	7	2.2963105722	1.798306
117	4	1	3	7	-2.3861761578	1.721078	118	4	4	3	7	-1.0311179996	1.736302
119	4	7	3	7	-0.4967310158	1.751418	120	4	10	3	7	-0.1561784673	1.762674
121	7	1	3	7	-3.3462160158	1.720004	122	7	4	3	7	-1.9785648333	1.726704
123	7	7	3	7	-1.4431644082	1.736302	124	7	10	3	7	-1.1073659233	1.745421
125	10	1	3	7	-3.9587968222	1.719728	126	10	4	3	7	-2.5841248138	1.723354
127	10	7	3	7	-2.0431671604	1.729507	128	10	10	3	7	-1.7057850005	1.736302
129	1	1	5	7	0.0003446144	1.251835	130	1	4	5	7	1.3868449453	1.257101
131	1	7	5	7	1.9465940171	1.257810	132	1	10	5	7	2.3033104371	1.258008
133	4	1	5	7	-1.7299433068	1.248610	134	4	4	5	7	-0.3487726767	1.251835
135	4	7	5	7	0.2071775518	1.254423	136	4	10	5	7	5.619829e-001	1.255849
137	7	1	5	7	-2.428145	1.248372	138	7	4	5	7	-1.045129	1.249847
139	7	7	5	7	-4.897035e-001	1.251835	140	7	10	5	7	-1.363346e-001	1.253487
141	10	1	5	7	-2.873247	1.248310	142	10	4	5	7	-1.489046	1.249114
143	10	7	5	7	-9.328103e-001	1.250448	144	10	10	5	7	-5.795267e-001	1.251835
145	1	1	7	7	-1.387102e-016	1	146	1	4	7	7	1.386294	1
147	1	7	7	7	1.945910	1	148	1	10	7	7	2.302585	1
149	4	1	7	7	-1.386294	1	150	4	4	7	7	-3.467754e-016	1
151	4	7	7	7	5.596158e-001	1	152	4	10	7	7	9.162907e-001	1
153	7	1	7	7	-1.945910	1	154	7	4	7	7	-5.596158e-001	1
155	7	7	7	7	2.774203e-016	1	156	7	10	7	7	3.566749e-001	1
157	10	1	7	7	-2.302585	1	158	10	4	7	7	-9.162907e-001	1
159	10	7	7	7	-3.566749e-001	1	160	10	10	7	7	-2.774203e-016	1
161	1	1	1	10	-2.239266e-001	3.987067	162	1	4	1	10	9.468466e-001	4.114020
163	1	7	1	10	1.44298765	4.185480	164	1	10	1	10	1.76888583	4.235675
165	4	1	1	10	-5.53040560	3.938556	166	4	4	1	10	-4.36488038	3.987067
167	4	7	1	10	-3.95522737	4.028775	168	4	10	1	10	-3.70170598	4.062278
169	7	1	1	10	-7.70650958	3.933820	170	7	4	1	10	-6.45977704	3.958874
171	7	7	1	10	-6.03649009	3.987067	172	7	10	1	10	-5.78711666	4.012086
173	10	1	1	10	-9.10136425	3.932527	174	10	4	1	10	-7.80656753	3.947447
175	10	7	1	10	-7.35939720	3.967572	176	10	10	1	10	-7.10190196	3.987067
177	1	1	4	10	-0.01356637	1.811878	178	1	4	4	10	1.36761087	1.858999
179	1	7	4	10	1.93214959	1.876756	180	1	10	4	10	2.29231173	1.855661

Columns	1	2	3	4	5	6	Columns	1	2	3	4	5	6
181	4	1	4	10	-2.48903668	1.794835	182	4	4	4	10	-1.13906879	1.811878
183	4	7	4	10	-0.60881952	1.828918	184	4	10	4	10	-0.27117863	1.841837
185	7	1	4	10	-3.48984092	1.793626	186	7	4	4	10	-2.12515395	1.801142
187	7	7	4	10	-1.59340875	1.811878	188	7	10	4	10	-1.26078985	1.822126
189	10	1	4	10	-4.12853468	1.793315	190	10	4	4	10	-2.755700	1.797390
191	10	7	4	10	-2.217544	1.804277	192	10	10	4	10	-1.882985	1.811878
193	1	1	7	10	2.532713e-004	1.268608	194	1	4	7	10	1.386799	1.274651
195	1	7	7	10	1.946585	1.275497	196	1	10	7	10	2.303313	1.275736
197	4	1	7	10	-1.752656	1.265015	198	4	4	7	10	-3.721166e-001	1.268608
199	4	7	7	10	1.833392e-001	1.271539	200	4	10	7	10	5.378646e-001	1.273182
201	7	1	7	10	-2.459975	1.264752	202	7	4	7	10	-1.077345	1.266388
203	7	7	7	10	-5.224339e-001	1.268608	204	7	10	7	10	-1.694983e-001	1.270472
205	10	1	7	10	-2.910901	1.264684	206	10	4	7	10	-1.526942	1.265574
207	10	7	7	10	-9.711066e-001	1.267058	208	10	10	7	10	-6.182396e-001	1.268608
209	1	1	10	10	-1.387102e-016	1	210	1	4	10	10	1.386294	1
211	1	7	10	10	1.945910	1	212	1	10	10	10	2.302585	1
213	4	1	10	10	-1.386294	1	214	4	4	10	10	-3.467754e-016	1
215	4	7	10	10	5.596158e-001	1	216	4	10	10	10	9.162907e-001	1
217	7	1	10	10	-1.945910	1	218	7	4	10	10	-5.596158e-001	1
219	7	7	10	10	2.774203e-016	1	220	7	10	10	10	3.566749e-001	1
221	10	1	10	10	-2.302585	1	222	10	4	10	10	-9.162907e-001	1
223	10	7	10	10	-3.566749e-001	1	224	10	10	10	10	-2.774203e-016	1

■

## Appendix F

# Finite Difference Algorithm:1

### Heading to a Minimal Point

The listings below are the files required to run the finite difference algorithm for the Boundary Value Problem. The solution to the Boundary Value Problem is equivalent to the solution of the Euler-Lagrange differential equations corresponding to the problem of finding a path on  $L(x, y)$  that minimises a line integral on  $L(x, y)$ . The files in this section finds the path when one of boundary points is a minimum on  $L(x, y)$

1)**optfinddiff.f**: this program finds the optimal path between  $x_0$  and the minimum of  $L(x, y) = a_1x^2 + a_3y^2 + a_6$  in order to minimise the path integral of  $L(x, y)$ . The assumption is that  $L(x, y)$  is a cup shaped surface.

2)**ff**: this program returns the value of  $f(x, y, y')$ . This function is the Euler-Lagrange equation in  $y''$

3)**ffy**: this program returns the value of  $f_y(x, y, y')$ . This function is the partial derivative of the Euler-Lagrange equation in  $y''$  with respect to  $y$ .

4)**fydash**: this program returns the value of  $f_{y'}(x, y, y')$ . This function is the partial derivative of the Euler-Lagrange equation in  $y''$  with respect to  $y'$ .

5)**transformout.f**: this program takes a generic cup shaped surface and transforms it to  $L(x, y) = a_1x^2 + a_3y^2 + a_6$ . It then calls *optfinddiff.f* to find the optimal path.

## The Function: `optfinddiff.f`

This program finds the path that minimises the line integral of  $L(x, y) = a_1x^2 + a_3y^2 + a_6$  by solving the corresponding Boundary Value Problem. The inputs are

### Input:

a : the x value of the initial position.

b : the x value of the final position.

alpha : the y value of the initial position.

beta : the y value of the final position.

N: the number of points in the path not counting the beginning and end points.

Toler: the error in successive approximations. When the algorithm gets to a value below Toler it ends the program and supplies the path found.

M: the number of iterations to do before failing without a solution.

a1-a6: the coefficients of  $L(x, y) = a_1x^2 + a_2x + a_3y^2 + a_4y + a_5xy + a_6$ . For this problem  $a_2, a_4$  and  $a_5$  are taken to be zero. If any of these are not zero then the function *transformout.f* should be called instead.

### Output:

As the program runs there will be output describing the iteration number, the tolerance level set as input and the current value of the tolerance. There will also be a graph displayed with the current path under consideration. This aids as a visual check to see that the algorithm is generating a smooth curve. If there are bumps in the curve which are increasing with each iteration it is likely to mean the algorithm will be unsuccessful in finding a solution.

The final output is a  $(N + 2) \times 3$  matrix. If the algorithm stops successfully the matrix will contain in the

first column: the x positions of the path

second column: the y positions of the path and

third column: a one in the first position to signify successful completion.

If the algorithm stops without success then the matrix will contain a set of zeros.

The algorithm is

```
optfinddiff.f<-
```



```
function(a, b, alpha, beta, N, Toler, M, a1, a2, a3, a4, a5, a6)
{
  plot(c(a, b), c(alpha, beta))
  xx <- matrix(0, N + 2, 3) #
#step 1
  h <- (b - a)/(N + 1)
  w <- matrix(0, N + 2, 1)
  w[1, 1] <- alpha
  w[N + 2, 1] <- beta #
#step 2
  if(a1 < a3) {
    grad <- exp(0.686696259 * log(a3/a1) - 0.035938571
               * log(a3/a1)^2)
  }
  if(a3 < a1) {
    grad <- 1/exp(0.686696259 * log(a1/a3) - 0.035938571
                 * log(a1/a3)^2)
  }
  if(a1 != a3) {
    for(i in 1:N) {
      w[i + 1, 1] <- alpha * ((a + i * h)/a)^(grad)
    }
  }
  if(a1 == a3) {
    for(i in 1:N) {
      w[i + 1, 1] <- alpha + (i * (beta - alpha))/(b - a) * h
    }
  }
  lines(seq(a, b, length = length(w)), w)
  points(seq(a, b, length = length(w)), w)
  k <- 1 #
#step 4
  aa <- matrix(0, N, 1)
  bb <- matrix(0, N, 1)
```

```

cc <- matrix(0, N, 1)
d <- matrix(0, N, 1)
l <- matrix(0, N, 1)
u <- matrix(0, N, 1)
v <- matrix(Toler, N, 1)
z <- matrix(0, N, 1)
while(k < M & sum(abs(v)) > Toler) {
#Step 5
  x <- a + h
  t <- (w[3, 1] - alpha)/(2 * h)
  aa[1, 1] <- 2+h^2*ff(y, w[2, 1], t, a1, a2, a3, a4, a5, a6)
  bb[1, 1] <- -1 + (h/2 * ff(ydash, w[2, 1], t, a1, a2, a3,
    a4, a5, a6))
  d[1, 1] <- - (2 * w[2, 1] - w[3, 1] - alpha + (h^2
    * ff(x, w[2, 1], t, a1, a2, a3, a4, a5, a6))) #
#step 6
  for(i in 2:(N - 1)) {
    x <- a + i * h
    t <- (w[i + 2, 1] - w[i, 1])/(2 * h)
    aa[i, 1] <- 2 + (h^2 * ff(y, w[i + 1, 1], t, a1, a2, a3,
      a4, a5, a6))
    bb[i, 1] <- -1 + (h/2 * ff(ydash, w[i + 1, 1], t, a1, a2,
      a3, a4, a5, a6))
    cc[i, 1] <- -1 - (h/2 * ff(ydash, w[i + 1, 1], t, a1, a2,
      a3, a4, a5, a6))
    d[i, 1] <- - (2 * w[i + 1, 1] - w[i + 2, 1] - w[i, 1] +
      (h^2 * ff(x, w[i + 1, 1], t, a1, a2, a3, a4, a5, a6)))
  }
#step 7
  x <- b - h
  t <- (beta - w[N, 1])/(2 * h)
  aa[N, 1] <- 2 + (h^2 * ff(y, w[N + 1, 1], t, a1, a2, a3, a4,
    a5, a6))
  cc[N, 1] <- -1 - (h/2 * ff(ydash, w[N + 1, 1], t, a1, a2, a3,

```

```

        a4, a5, a6))
d[N, 1] <- - (2 * w[N + 1, 1] - beta - w[N, 1] + (h^2 *
        ff(x, w[N + 1, 1], t, a1, a2, a3, a4, a5, a6))) #
# step 8
l[1, 1] <- aa[1, 1]
u[1, 1] <- bb[1, 1]/aa[1, 1] #
#step 9
for(i in 2:(N - 1)) {
    l[i, 1] <- aa[i, 1] - cc[i, 1] * u[i - 1, 1]
    u[i, 1] <- bb[i, 1]/l[i, 1]
}
#step 10
l[N, 1] <- aa[N, 1] - cc[N, 1] * u[N - 1, 1] #
#step 11
z[1, 1] <- d[1, 1]/l[1, 1] #
#step 12
for(i in 2:N) {
    z[i, 1] <- (d[i, 1] - cc[i, 1] * z[i - 1, 1])/l[i, 1]
}
#step 13
v[N, 1] <- z[N, 1]
w[N + 1, 1] <- w[N + 1, 1] + v[N, 1] #
#step 14
for(i in (N - 1):1) {
    v[i, 1] <- z[i, 1] - u[i, 1] * v[i + 1, 1]
    w[i + 1, 1] <- w[i + 1, 1] + v[i, 1]
}
#step 15
#step 16
cat(k, Toler, sum(abs(v)), N, fill = T)
if(sum(abs(v)) < Toler) {
    for(i in 1:(N + 2)) {
        xx[i, 1] <- a + (i - 1) * h
    }
}

```

```

# step 17
#STOP
      xx[, 2] <- w[, 1]
      xx[1, 3] <- 1
      cat("Done", fill = T)
    }
#step 18
#
      lines(seq(a, b, length = length(w)), w)
      points(seq(a, b, length = length(w)), w)
      k <- k + 1
    }
#step 19
  xx
}

```

### The Function:ff

This program returns the value of  $f(x, y, y')$ .

```

ff<-
function(x, y, ydash, a, b, c, d, e, f)
{
  ((1 + ydash^2) * ((2 * c * y + d + e * x) - (2 * a * x + b + e * y)
    * ydash))/(a * x^2 + b * x + c * y^2 + d * y + e * x * y + f)
}

```

### The Function:ffy

This program returns the value of  $f_y(x, y, y')$ .

```

ffy<-
function(x, y, ydash, a, b, c, d, e, f)
{
  (1 + ydash^2) * ((2 * c - e * ydash)/(a * x^2 + b * x + c * y^2
    + d * y + e * x * y + f) - ((2 * c * y + d + e * x) *
    ((2 * c * y + d + e * x) - (2 * x * a +

```

```

    b + e * y) * ydash))/(a * x^2 + b * x + c * y^2 + d *
    y + e * x * y + f)^2)
}

```

### The Function:ffydash

This program returns the value of  $f_{y'}(x, y, y')$

```

ffydash<-
function(x, y, ydash, a, b, c, d, e, f)
{
    (2 * ydash * ((2 * c * y + d + e * x) - (2 * a * x + b + e * y)
    * ydash)) / (a * x^2 + b * x + c * y^2 + d * y + e * x
    * y + f) - ((2 * a * x + b + e * y) * (1 + ydash^2)) /
    (a * x^2 + b * x + c * y^2 + d * y + e * x * y + f)
}

```

### The Function:transformout.f

This program finds the optimal path that minimises the line integral for the surface  $L(x, y) = a_1x^2 + a_2x + a_3y^2 + a_4y + a_5xy + a_6$ . The inputs are

#### Input:

a : the x value of the initial position.

b : the x value of the final position.

alpha : the y value of the initial position.

beta : the y value of the final position.

N: the number of points in the path not counting the beginning and end points.

Toler: the error in successive approximations. When the algorithm gets to a value below Toler it ends the program and supplies the path found.

M: the number of iterations to do before failing.

a1-a6: the coefficients of  $L(x, y) = a_1x^2 + a_2x + a_3y^2 + a_4y + a_5xy + a_6$ .

#### Output:

As the program runs there will be output describing the iteration number, the tolerance level set as input and the current value of the tolerance. There will also

be a graph displayed with the current path under consideration. This aids as a visual check to see that the algorithm is generating a smooth curve. If there are bumps in the curve which are increasing with each iteration it is likely to mean the algorithm will be unsuccessful. The algorithm works on the transformed data and the labels on the graph refer to the transformed data values. The final output is back transformed and can be graphed.

The final output is a  $(N + 2) \times 3$  matrix. If the algorithm stop successfully the matrix will contain in the

first column: the x positions of the path

second column: the y positions of the path and

third column: a one in the first position to signify successful completion.

If the algorithm stops without success then the matrix will contain a set of zeros.

The algorithm is

transformout.f

```
function(a, b, alpha, beta, N, Toler, M, a1, a2, a3, a4, a5, a6)
```

```
{
```

```
  b1 <- a1
```

```
  b2 <- a2
```

```
  b3 <- a3
```

```
  b4 <- a4
```

```
  b5 <- a5
```

```
  b6 <- a6
```

```
  a0 <- a
```

```
  b0 <- b
```

```
  alpha0 <- alpha
```

```
  beta0 <- beta
```

```
  if(a5 != 0) {
```

```
#
```

```
#find angle of rotation
```

```
  theta <- 1/2 * atan(a5/(a1 - a3)) #
```

```
#recovers if (a1-a3) = 0 !
```

```
#find new coeffs of equation
```

```
  b1 <- a1 * (cos(theta))^2 + a5 * cos(theta) * sin(theta)
    + a3 * (sin(theta))^2
```

```

    b3 <- a1 * (sin(theta))^2 - a5 * sin(theta) * cos(theta)
      + a3 * (cos(theta))^2
    b2 <- a2 * cos(theta) + a4 * sin(theta)
    b4 <- -a2 * sin(theta) + a4 * cos(theta)
    b5 <- a5 * ((cos(theta))^2 - (sin(theta))^2) +
      2 * ((a3 - a1)) * sin(theta) * cos(theta)
    b6 <- a6 #
#finds new alpha, beta, a0 and b0
    a0 <- a * cos(theta) + alpha * sin(theta)
    alpha0 <- - a * sin(theta) + alpha * cos(theta)
    b0 <- b * cos(theta) + beta * sin(theta)
    beta0 <- - b * sin(theta) + beta * cos(theta)
  } #
#adjusts the axes so minimum is at (0,0)
    a0 <- a0 + (b2/(2 * b1))
    alpha0 <- alpha0 + (b4/(2 * b3))
    b0 <- b0 + (b2/(2 * b1))
    beta0 <- beta0 + (b4/(2 * b3)) #
# check to see which way optdiff is more likely to work
    if(b1 <= b3) {
      getx <-- optfinddiff.f(a0, b0, alpha0, beta0, N, Toler,
        M, b1, 0, b3, 0, 0, b6)
    }
    if(b3 < b1) {
      getx <-- optfinddiff.f(alpha0, beta0, a0, b0, N, Toler,
        M, b3, 0, b1, 0, 0, b6)
      hold <- getx[, 1]
      getx[, 1] <- getx[, 2]
      getx[, 2] <- hold
    }
# re-adjusts the minimum
    gx <- getx[, 1] - (b2/(2 * b1))
    gy <- getx[, 2] - (b4/(2 * b3)) #
#return solution to the old co-ordinates

```

```

if(a5 != 0) {
  ggx <- gx * cos(theta) - gy * sin(theta)
  ggy <- gx * sin(theta) + gy * cos(theta)
  gx <- ggx
  gy <- ggy
}
cbind(gx, gy, getx[,3])
}

```

## Typical Input

This is the Splus command to call *transformout.f*. The starting position is (3, 1) and the end position is (0, 0). There are to be 20 points on the path (besides the points (3, 1) and (0, 0)). The tolerance is 0.000001 and 100 iterations will be considered. The value of  $L$  is  $L(x, y) = 1x^2 + 100y^2 - 1xy$ .

```
gg<-transformout.f(3,0,1,0,20,0.000001,100,1,0,100,0,-1,0)
```

## Typical Output

The screen output is

```

1 1e-006 0.76853690975329 20
2 1e-006 0.0770720818316491 20
3 1e-006 0.00837555031841968 20
4 1e-006 0.000100336134869916 20
5 1e-006 1.62367731103413e-008 20

```

The path data is

```

gg
      gx      gy
[1,] 3.0000000 1.0000000000
[2,] 2.8602917 0.3288987357
[3,] 2.7180483 0.1597504459
[4,] 2.5753417 0.0823193350
[5,] 2.4324357 0.0443803052
[6,] 2.2894317 0.0258368226

```



```
[7,] 2.1463793 0.0168871458
[8,] 2.0033036 0.0125404514
[9,] 1.8602172 0.0103060083
[10,] 1.7171262 0.0089908113
[11,] 1.5740333 0.0080519528
[12,] 1.4309397 0.0072566798
[13,] 1.2878458 0.0065118780
[14,] 1.1447519 0.0057831887
[15,] 1.0016579 0.0050590871
[16,] 0.8585639 0.0043361236
[17,] 0.7154699 0.0036133987
[18,] 0.5723759 0.0028907144
[19,] 0.4292820 0.0021680354
[20,] 0.2861880 0.0014453569
[21,] 0.1430940 0.0007226785
[22,] 0.0000000 0.0000000000
```

Finally here is the graph of the optimal path found by *transformout.f*

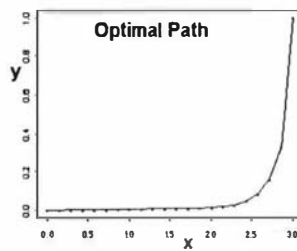


Figure F.1: The graph of the path found by *transformout.f*

■

## Appendix G

# Finite Difference Algorithm:2

### Heading to a Non-Minimal Point

The listings below are the files required to run the finite difference algorithm for the Boundary Value Problem. The solution to the Boundary Value Problem is equivalent to the solution of the Euler-Lagrange differential equations corresponding to the problem of finding a path on  $L(x, y)$  that minimises the line integral on  $L(x, y)$ . The files in this section find the solution when neither boundary points is a minimum on  $L(x, y)$

1) **foptfinddiff.f**: this program finds the optimal path between  $x_0$  and  $x_1$  for the surface  $L(x, y) = a_1x^2 + a_3y^2 + a_6$  in order to minimise the path integral of  $L(x, y)$ . The assumption is that  $L(x, y)$  is a cup shaped surface.

2) **ff**: this program returns the value of  $f(x, y, y')$ . This function is the Euler-Lagrange equation in  $y''$

3) **fy**: this program returns the value of  $f_y(x, y, y')$ . This function is the partial derivative of the Euler-Lagrange equation in  $y''$  with respect to  $y$ .

4) **fydash**: this program returns the value of  $f_{y'}(x, y, y')$ . This function is the partial derivative of the Euler-Lagrange equation in  $y''$  with respect to  $y'$ .

5) **transformoutnomin.f**: this program takes a generic cup shaped surface, a polynomial of order 2 in two variables, and transforms it to  $L(x, y) = a_1x^2 + a_3y^2 + a_6$ . It then calls *foptfinddiff.f* to find the optimal path.

## The Function:foptfinddiff.f

This program finds the path that minimises the line integral of  $L(x, y) = a_1x^2 + a_3y^2 + a_6$  by solving the corresponding Boundary Value Problem. The inputs are

### Input:

a : the x value of the initial position.

b : the x value of the final position.

alpha : the y value of the initial position.

beta : the y value of the final position.

N: the number of points on the path not counting the beginning and end points.

Toler: the error in successive approximations. When the algorithm gets to a value below Toler it ends the program and supplies the path found.

M: the number of iterations to do before failing without a solution.

a1-a6: the coefficients of  $L(x, y) = a_1x^2 + a_2x + a_3y^2 + a_4y + a_5xy + a_6$ . For this problem  $a_2, a_4$  and  $a_5$  are taken to be zero. If any of these are not then the function *transformout.f* should be called instead.

### Output:

As the program runs there will be output describing the iteration number, the tolerance level set as input and the current value of the tolerance. There will also be a graph displayed with the current path under consideration. This aids as a visual check to see that the algorithm is generating a smooth curve. If there are bumps in the curve which are increasing with each iteration it is likely to mean the algorithm will be unsuccessful.

The final output is a  $(N + 2) \times 3$  matrix. If the algorithm stops successfully the matrix will contain in the

first column: the x positions of the path

second column: the y positions of the path and

third column: a one in the first position to signify successful completion.

If the algorithm stops without success then the matrix will contain a set of zeros.

The algorithm is

```
foptfinddiff.f<-
```

```

function(a, b, alpha, beta, N, Toler, M, a1, a2, a3, a4, a5, a6)
{
  plot(c(a, b), c(alpha + 1, beta))
  xx <- matrix(0, N + 2, 3) #
#step 1
  h <- (b - a)/(N + 1)
  w <- matrix(0, N + 2, 1)
  w[1, 1] <- alpha
  w[N + 2, 1] <- beta #
#step 2
  for(i in 1:N) {
    w[i + 1, 1] <- alpha + (i * (beta - alpha))/(b - a) * h
  }
  lines(seq(a, b, length = length(w)), w)
  points(seq(a, b, length = length(w)), w)
  k <- 1 #
#step 4
  aa <- matrix(0, N, 1)
  bb <- matrix(0, N, 1)
  cc <- matrix(0, N, 1)
  d <- matrix(0, N, 1)
  l <- matrix(0, N, 1)
  u <- matrix(0, N, 1)
  v <- matrix(Toler, N, 1)
  z <- matrix(0, N, 1)
  while(k < M & sum(abs(v)) > Toler) {
#Step 5
    x <- a + h
    t <- (w[3, 1] - alpha)/(2 * h)
    aa[1, 1] <- 2 + h^2 * ffy(x, w[2, 1], t, a1, a2, a3, a4,
      a5, a6)
    bb[1, 1] <- -1 + (h/2 * ffydash(x, w[2, 1], t, a1, a2, a3,
      a4, a5, a6))
    d[1, 1] <- - (2 * w[2, 1] - w[3, 1] - alpha + (h^2 * ff(x,

```

```

        w[2, 1], t, a1, a2, a3, a4, a5, a6))) #
#step 6
for(i in 2:(N - 1)) {
  x <- a + i * h
  t <- (w[i + 2, 1] - w[i, 1])/(2 * h)
  aa[i, 1] <- 2 + (h^2 * ffy(x, w[i + 1, 1], t, a1, a2, a3,
    a4, a5, a6))
  bb[i, 1] <- -1 + (h/2 * ffydash(x, w[i + 1, 1], t, a1, a2,
    a3, a4, a5, a6))
  cc[i, 1] <- -1 - (h/2 * ffydash(x, w[i + 1, 1], t, a1, a2,
    a3, a4, a5, a6))
  d[i, 1] <- - (2 * w[i + 1, 1] - w[i + 2, 1] - w[i, 1] +
    (h^2 * ff(x, w[i + 1, 1], t, a1, a2, a3, a4, a5, a6)))
}
#step 7
x <- b - h
t <- (beta - w[N, 1])/(2 * h)
aa[N, 1] <- 2 + (h^2 * ffy(x, w[N + 1, 1], t, a1, a2, a3, a4,
  a5, a6))
cc[N, 1] <- -1 - (h/2 * ffydash(x, w[N + 1, 1], t, a1, a2, a3,
  a4, a5, a6))
d[N, 1] <- - (2 * w[N + 1, 1] - beta - w[N, 1] + (h^2 * ff(x,
  w[N + 1, 1], t, a1, a2, a3, a4, a5, a6))) #
# step 8
l[1, 1] <- aa[1, 1]
u[1, 1] <- bb[1, 1]/aa[1, 1] #
#step 9
for(i in 2:(N - 1)) {
  l[i, 1] <- aa[i, 1] - cc[i, 1] * u[i - 1, 1]
  u[i, 1] <- bb[i, 1]/l[i, 1]
}
#step 10
l[N, 1] <- aa[N, 1] - cc[N, 1] * u[N - 1, 1] #
#step 11

```

```
        z[1, 1] <- d[1, 1]/l[1, 1] #
#step 12
        for(i in 2:N) {
            z[i, 1] <- (d[i, 1] - cc[i, 1] * z[i - 1, 1])/l[i, 1]
        }
#step 13
        v[N, 1] <- z[N, 1]
        w[N + 1, 1] <- w[N + 1, 1] + v[N, 1] #
#step 14
        for(i in (N - 1):1) {
            v[i, 1] <- z[i, 1] - u[i, 1] * v[i + 1, 1]
            w[i + 1, 1] <- w[i + 1, 1] + v[i, 1]
        }
#step 15
#step 16
        cat(k, Toler, sum(abs(v)), N, fill = T)
        if(sum(abs(v)) < Toler) {
            for(i in 1:(N + 2)) {
                xx[i, 1] <- a + (i - 1) * h
            }
# step 17
#STOP
            xx[, 2] <- w[, 1]
            xx[1, 3] <- 1
            cat("Done", fill = T)
        }
#step 18
        lines(seq(a, b, length = length(w)), w)
        points(seq(a, b, length = length(w)), w)
        k <- k + 1
    }
    xx
}
```

**The Function:ff**

This program returns the value of  $f(x, y, y')$ .

```
ff<-function(x, y, ydash, a, b, c, d, e, f)
{
  ((1 + ydash^2) * ((2 * c * y + d + e * x) - (2 * a * x + b + e * y)
    * ydash))/(a * x^2 + b * x + c * y^2 + d * y + e * x * y + f)
}
```

**The Function:ffy**

This program returns the value of  $f_y(x, y, y')$ .

```
ffy<-function(x, y, ydash, a, b, c, d, e, f)
{
  (1 + ydash^2) * ((2 * c - e * ydash)/(a * x^2 + b * x + c * y^2
    + d * y + e * x * y + f) - ((2 * c * y + d + e * x) *
    ((2 * c * y + d + e * x) - (2 * x * a +
    b + e * y) * ydash))/(a * x^2 + b * x + c * y^2 + d *
    y + e * x * y + f)^2)
}
```

**The Function:ffydash**

This program returns the value of  $f_{y'}(x, y, y')$

```
ffydash <- function(x, y, ydash, a, b, c, d, e, f)
{
  (2 * ydash * ((2 * c * y + d + e * x) - (2 * a * x + b + e * y)
    * ydash)) / (a * x^2 + b * x + c * y^2 + d * y + e * x
    * y + f) - ((2 * a * x + b + e * y) * (1 + ydash^2)) /
    (a * x^2 + b * x + c * y^2 + d * y + e * x * y + f)
}
```

**The Function:transformoutnomin.f**

This program finds the optimal path that minimises the line integral for the surface  $L(x, y) = a_1x^2 + a_2x + a_3y^2 + a_4y + a_5xy + a_6$ . The inputs are

**Input:**

a : the x value of the initial position.

b : the x value of the final position.

alpha : the y value of the initial position.

beta : the y value of the final position.

N: the number of points in the path not counting the beginning and end points.

Toler: the error in successive approximations. When the algorithm gets to a value below Toler it ends the program and supplies the path found.

M: the number of iterations to do before failing.

a1-a6: the coefficients of  $L(x, y) = a_1x^2 + a_2x + a_3y^2 + a_4y + a_5xy + a_6$ .

**Output:**

As the program runs there will be output describing the iteration number, the tolerance level set as input and the current value of the tolerance. There will also be a graph displayed with the current path under consideration. This aids as a visual check to see that the algorithm is generating a smooth curve. If there are bumps in the curve which are increasing with each iteration it is likely to mean the algorithm will be unsuccessful. The algorithm works on the transformed data so, the labels on the graph refer to the transformed data values. The final output is back transformed and can be graphed.

The final output is a  $(N + 2) \times 3$  matrix. If the algorithm stop successfully the matrix will contain in the

first column: the x positions of the path

second column: the y positions of the path and

third column: a one in the first position to signify successful completion.

If the algorithm stops without success then the matrix will contain a set of zeros.

The algorithm is

```
transformoutnomin.f
```

```
function(a, b, alpha, beta, N, Toler, M, a1, a2, a3, a4, a5, a6)
```

```
{
```

```
  b1 <- a1
```

```
  b2 <- a2
```

```
  b3 <- a3
```



```

b4 <- a4
b5 <- a5
b6 <- a6
a0 <- a
b0 <- b
alpha0 <- alpha
beta0 <- beta
if(a5 != 0) {
#
#find angle of rotation
  theta <- 1/2 * atan(a5/(a1 - a3)) #
#recovers if (a1-a3) = 0 !
#find new coeffs of equation
  b1 <- a1 * (cos(theta))^2 + a5 * cos(theta) * sin(theta)
    + a3 * (sin(theta))^2
  b3 <- a1 * (sin(theta))^2 - a5 * sin(theta) * cos(theta)
    + a3 * (cos(theta))^2
  b2 <- a2 * cos(theta) + a4 * sin(theta)
  b4 <- - a2 * sin(theta) + a4 * cos(theta)
  b5 <- a5 * ((cos(theta))^2 - (sin(theta))^2)
    + 2 * ((a3 - a1)) * sin(theta) * cos(theta)
  b6 <- a6 #
#finds new alpha, beta, a0 and b0
  a0 <- a * cos(theta) + alpha * sin(theta)
  alpha0 <- - a * sin(theta) + alpha * cos(theta)
  b0 <- b * cos(theta) + beta * sin(theta)
  beta0 <- - b * sin(theta) + beta * cos(theta)
}
a0 <- a0 + (b2/(2 * b1))
alpha0 <- alpha0 + (b4/(2 * b3))
b0 <- b0 + (b2/(2 * b1))
beta0 <- beta0 + (b4/(2 * b3))
# check to see which way optdiff is more likely to work
getx <<- foptfinddiff.f(a0, b0, alpha0, beta0, N, Toler, M,

```

```

        b1, 0, b3, 0, 0, b6)
  gx <- getx[, 1] - (b2/(2 * b1))
  gy <- getx[, 2] - (b4/(2 * b3)) #
#return solution to the old co-ordinates
  if(a5 != 0) {
    ggx <- gx * cos(theta) - gy * sin(theta)
    ggy <- gx * sin(theta) + gy * cos(theta)
    gx <- ggx
    gy <- ggy
  }
  cbind(gx, gy, getx[,3])
}

```

## Typical Input

This is the Splus command to call *transformoutnomi.f*. The starting position is (3, 1) and the end position is (1, 1). There are to be 40 points on the path (besides the points (3, 1) and (1, 1)). The tolerance is 0.000001 and 100 iterations will be considered. The value of  $L$  is  $L(x, y) = 1x^2 + 1y^2 - 1xy$ .

```
g2<-transformoutnomi.f(3,1,1,1,40,0.000001,100,1,0,1,0,-1,0)
```

## Typical Output

The screen output is

```

1 1e-006 1.00775265276267 40
2 1e-006 0.05659800861419 40
3 1e-006 0.000120086434477465 40
4 1e-006 3.2598197324736e-010 40

```

The path data is

```

g2
      gx      gy
[1,] 3.000000 1.000000
[2,] 2.953246 1.0020268

```

[3,]	2.906258	1.0038194
[4,]	2.859032	1.0053731
[5,]	2.811561	1.0066832
[6,]	2.763843	1.0077452
[7,]	2.715872	1.0085546
[8,]	2.667644	1.0091073
[9,]	2.619155	1.0093994
[10,]	2.570403	1.0094276
[11,]	2.521384	1.0091890
[12,]	2.472096	1.0086815
[13,]	2.422538	1.0079038
[14,]	2.372710	1.0068559
[15,]	2.322612	1.0055389
[16,]	2.272248	1.0039557
[17,]	2.221624	1.0021113
[18,]	2.170745	1.0000132
[19,]	2.119623	0.9976719
[20,]	2.068272	0.9951015
[21,]	2.016711	0.9923205
[22,]	1.964962	0.9893524
[23,]	1.913056	0.9862270
[24,]	1.861030	0.9829809
[25,]	1.808927	0.9796591
[26,]	1.756803	0.9763155
[27,]	1.704722	0.9730146
	gx	gy
[28,]	1.652759	0.9698319
[29,]	1.601001	0.9668546
[30,]	1.549548	0.9641822
[31,]	1.498511	0.9619255
[32,]	1.448011	0.9602057
[33,]	1.398177	0.9591523
[34,]	1.349144	0.9588996
[35,]	1.301046	0.9595830

[36,] 1.254017 0.9613337  
[37,] 1.208177 0.9642746  
[38,] 1.163637 0.9685149  
[39,] 1.120488 0.9741463  
[40,] 1.078802 0.9812409  
[41,] 1.038629 0.9898490  
[42,] 1.000000 1.0000000

The graph of the solutions is in figure G.1.

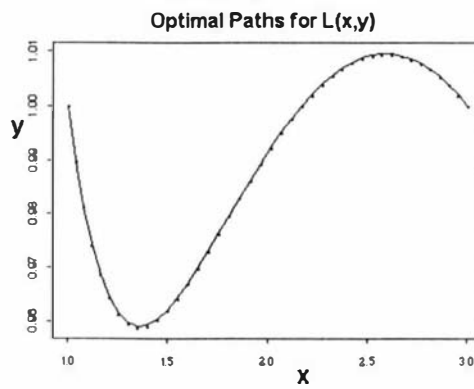


Figure G.1: The optimal path on  $L = x^2 + y^2 - xy$  from (3,1) to (1,1).

■

## Appendix H

# Design Matrix for the Split Plot Data

The data in the table below is that collected for the split plot example in Chapter 1.

shape	tar	coke	volt	cont	temp	data
-1	-1	-1	-1	-1	-1	1.15875977
1	-1	-1	-1	-1	-1	0.20568296
-1	1	-1	-1	-1	-1	1.51059135
1	1	-1	-1	-1	-1	-1.00010966
-1	-1	1	-1	-1	-1	0.76191541
1	-1	1	-1	-1	-1	2.70721242
-1	1	1	-1	-1	-1	0.76840067
1	1	1	-1	-1	-1	-0.13835194
-1	-1	-1	1	-1	-1	-0.32344358
1	-1	-1	1	-1	-1	-1.95964132
-1	1	-1	1	-1	-1	0.79212294
1	1	-1	1	-1	-1	-1.28714965
-1	-1	1	1	-1	-1	0.14217644
1	-1	1	1	-1	-1	-0.48589399
-1	1	1	1	-1	-1	-0.06517826
1	1	1	1	-1	-1	1.03065844
-1	-1	-1	-1	1	-1	-2.09651132
1	-1	-1	-1	1	-1	-0.09722972
-1	1	-1	-1	1	-1	0.16299795

1	1	-1	-1	1	-1	0.82381815
-1	-1	1	-1	1	-1	-1.54259351
1	-1	1	-1	1	-1	-0.34825481
-1	1	1	-1	1	-1	0.25956763
1	1	1	-1	1	-1	1.39071878
-1	-1	-1	1	1	-1	-0.36839460
1	-1	-1	1	1	-1	-0.65586610
-1	1	-1	1	1	-1	-0.02044628
1	1	-1	1	1	-1	0.02623366
-1	-1	1	1	1	-1	2.18119632
1	-1	1	1	1	-1	-1.25550984
-1	1	1	1	1	-1	-0.09207472
1	1	1	1	1	-1	0.63682325
-1	-1	-1	-1	-1	1	2.09393221
1	-1	-1	-1	-1	1	-1.51767761
-1	1	-1	-1	-1	1	0.48643948
1	1	-1	-1	-1	1	0.71073556
-1	-1	1	-1	-1	1	-1.29747482
1	-1	1	-1	-1	1	-0.63506206
-1	1	1	-1	-1	1	0.15379463
1	1	1	-1	-1	1	-1.24940022
-1	-1	-1	1	-1	1	-1.38599347
1	-1	-1	1	-1	1	1.01398356
-1	1	-1	1	-1	1	-1.43891814
1	1	-1	1	-1	1	-0.12583428
-1	-1	1	1	-1	1	0.68550895
1	-1	1	1	-1	1	0.59457865
-1	1	1	1	-1	1	-1.16981467
1	1	1	1	-1	1	-0.6197410
-1	-1	-1	-1	1	1	0.3829985
1	-1	-1	-1	1	1	-1.5483784
-1	1	-1	-1	1	1	2.0846723
1	1	-1	-1	1	1	-0.5608030
-1	-1	1	-1	1	1	0.7745685
1	-1	1	-1	1	1	-0.1349605
-1	1	1	-1	1	1	-1.3571536
1	1	1	-1	1	1	-1.8949398

-1	-1	-1	1	1	1	1.0059796
1	-1	-1	1	1	1	-0.3897107
-1	1	-1	1	1	1	0.2757704
1	1	-1	1	1	1	0.1216626
-1	-1	1	1	1	1	-1.8323588
1	-1	1	1	1	1	-0.4633447
-1	1	1	1	1	1	1.9977313
1	1	1	1	1	1	0.8531242



# Appendix I

## Determining Minimal Transitions:1

### One Factor per Change

This algorithm *whichtochange.f* finds the transition from  $\mathbf{x}_0$  to  $\mathbf{x}_1$  that has the smallest value. A transition value is the sum of the line integrals on  $L(\mathbf{x})$  corresponding to each factor change.  $L(\mathbf{x})$  is a variance or an adjusted variance function. It is a polynomial of order 2 in  $p$  controllable factors and can be represented by

$$L(\mathbf{x}) = A + \mathbf{x}^t \mathbf{B} + \mathbf{x}^t \mathbf{C} \mathbf{x}$$

Where  $\mathbf{C}$  is a symmetric matrix.

### **whichtochange.f**

#### **Inputs**

The inputs are

**paths:** This is a matrix of dimensions  $n_t \times p$ . Where  $p$  is the number of controllable factors and  $n_t$  is the number of paths to be considered. A row in the path matrix represents a transition and is the order that the  $p$  factors are changed e.g a typical row for a 3 factor problem would be (2,1,3) which means the second factor is changed first, then the first factor and lastly the third. The largest values that  $n_t$  can take without there being repeats is  $p!$ . Some transitions may not be practical and these need not be listed in the paths matrix.



**AA:** represents the constant,  $A$ , in the equation  $L(\mathbf{x}) = A + \mathbf{x}'\mathbf{B} + \mathbf{x}'\mathbf{C}\mathbf{x}$ .

**BB:** represents the linear terms,  $\mathbf{B}$ , in the equation  $L(\mathbf{x}) = A + \mathbf{x}'\mathbf{B} + \mathbf{x}'\mathbf{C}\mathbf{x}$ . It is a  $p \times 1$  matrix.

**CC:** represents the interaction and squared terms,  $\mathbf{C}$ , in the equation  $L(\mathbf{x}) = A + \mathbf{x}'\mathbf{B} + \mathbf{x}'\mathbf{C}\mathbf{x}$ .  $\mathbf{C}$  is a  $p \times p$  symmetric matrix where each coefficient of an interaction will appear twice and so is halved.

**start:** represents the  $p \times 1$  matrix  $\mathbf{x}_0$

**finish:** represents the  $p \times 1$  matrix  $\mathbf{x}_1$

## Outputs

As the algorithm proceeds it outputs the row number of the transition under consideration and the sum of the line integrals for the paths in that transition. The final output comprises a list of references to the rows of the matrix, *paths* that correspond to transitions with the smallest value i.e. the output 4 7 would mean that there are two transitions that have the smallest value and that the transition order is contained in rows 4 and rows 7 of the matrix *paths*. The other object is the value of the sum of the line integrals for the paths that have minimal transitions.

The algorithm is

```
whichtochange.f<-
function(paths, AA, BB, CC, start, finish)
{
  noofpaths <- dim(paths)[1]
  nooffactors <- dim(paths)[2]
  minpath <- c()
  minpathlength <- 1000000000000
  currentpathlength <- 0
  for(i in 1:noofpaths) {
    currentpos <- start
    currentpathlength <- 0
    for(j in 1:nooffactors) {
      currentpathlength <- currentpathlength + pathintegral(paths[i,
        j], currentpos, AA, BB, CC, finish[paths[i, j]], start[
        paths[i, j]])
    }
  }
}
```

```

        currentpos[paths[i, j]] <- finish[paths[i, j]]
    }
    cat(i, currentpathlength, fill = T)
    if(currentpathlength == minpathlength) {
        minpath <- c(minpath, i)
    }
    if(currentpathlength < minpathlength) {
        minpath <- i
        minpathlength <- currentpathlength
    }
}
list(paths = minpath, min.path.length = minpathlength)
}

```

### pathintegral

This function is called by *whichtochange.f* and is a generic solver for line integrals of the type specified by  $L(x)$  when only one variable is changing.

```

pathintegral<-
function(x, values, AA, BB, CC, lower, upper)
{
    BBB <- BB
    BBB[x, 1] <- BB[x, 1]/2
    CCC <- CC
    CCC[, x] <- CC[, x]/2
    CCC[x, ] <- CCC[x, ]/2
    CCC[x, x] <- CC[x, x]/3
    uppervalues <- values
    uppervalues[x, 1] <- upper
    uppervalues <- matrix(uppervalues, , 1)
    lowervalues <- values
    lowervalues[x, 1] <- lower
    lowervalues <- matrix(lowervalues, , 1)
    abs(upper * (AA + t(uppervalues) %*% BBB + t(uppervalues) %*%
        CCC %*% uppervalues) - lower * (AA + t(lowervalues) %*% BBB

```

```

+ t(lowervalues) %% CCC %% lowervalues))
}

```

## Typical Input

Suppose  $L(x, y, z) = 10 + x + 3y + 4z + 5x^2 + 6y^2 + 3z^2 + 4xy + 2xz + 6yz$  then  $L(x, y, z)$  can be represented by

$$L(x, y, z) = 10 + (x, y, z) \begin{pmatrix} 1 \\ 3 \\ 4 \end{pmatrix} + (x, y, z) \begin{pmatrix} 5 & 2 & 1 \\ 2 & 6 & 3 \\ 1 & 3 & 3 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix}$$

i.e.

$$A = 10, B = \begin{pmatrix} 1 \\ 3 \\ 4 \end{pmatrix}, C = \begin{pmatrix} 5 & 2 & 1 \\ 2 & 6 & 3 \\ 1 & 3 & 3 \end{pmatrix}$$

As input  $AA = A$ ,  $BB = B$  and  $CC = C$ .

The initial position is  $\mathbf{x}_0 = \begin{pmatrix} 3 \\ 2 \\ 4 \end{pmatrix}$  and input as the variable *start*.

The final position is  $\mathbf{x}_1 = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$  and is input as the variable *finish*.

The last input is the *paths* matrix. There are three factors and so there are  $3! = 6$  unique transitions. The matrix *paths* below represents those transitions.

1	2	3
1	3	2
2	1	3
2	3	1
3	2	1
3	1	2

i.e. in row 6 the transition is to move factor 3,  $z$ , first, then move factor 1,  $x$ , and then move factor 2,  $y$ .

## Typical Output

The function call is *whichtochange.f(paths, AA, BB, CC, start, finish)*. The output is

1 918.5

2 948.5

3 816.5

4 807.5

5 873.5

6 903.5

\$paths:

[1] 4

\$min.path.length:

[,1]

[1,] 807.5

The transition with the lowest sum of path integrals is in row 4 and the sum is 807.5. This transition is (2, 3, 1) i.e. *y* is moved first, then *z* and then *x*.

## Calculations By Hand

Calculating the first transition, (1, 2, 3), by hand, as a final check gives

Path 1.

The first factor is changed from value 3 to 0 with factors 2 and 3 remaining constant at 2 and 4 respectively.

$$\begin{aligned}
 & \int_0^3 10 + x + 3y + 4z + 5x^2 + 6y^2 + 3z^2 + 4xy + 2xz + 6yz \, |dx| \\
 = & \left[ 10x + \frac{x^2}{2} + 3 \times 2x + 4 \times 4x + \frac{5x^3}{3} + 6 \times (2)^2x + 3 \times (4)^2x \right. \\
 & \left. + \frac{4x^2(2)}{2} + \frac{2x^2 \times 4}{2} + 6 \times 2 \times 4x \right]_0^3 \\
 = & 30 + 4.5 + 18 + 48 + 45 + 72 + 144 + 36 + 36 + 144 \\
 = & 577.5
 \end{aligned}$$

Path 2.

The second factor is changed from value 2 to 0 with factors 1 and 3 remaining

constant at 0 and 4 respectively.

$$\begin{aligned}
 & \int_0^2 10 + x + 3y + 4z + 5x^2 + 6y^2 + 3z^2 + 4xy + 2xz + 6yz \, |dy| \\
 &= \left[ 10y + \frac{3y^2}{2} + 4 \times 4y + \frac{6y^3}{3} + 3 \times 4^2y + \frac{6y^2}{2} \right]_0^2 \\
 &= 20 + 6 + 32 + 16 + 96 + 48 \\
 &= 218
 \end{aligned}$$

Path 3.

The third factor is changed from value 4 to 1 with factors 1 and 2 both remaining constant at 0.

$$\begin{aligned}
 & \int_1^4 10 + x + 3y + 4z + 5x^2 + 6y^2 + 3z^2 + 4xy + 2xz + 6yz \, |dz| \\
 &= \left[ 10z + \frac{4z^2}{2} + \frac{3z^3}{3} \right]_1^4 \\
 &= 40 + 32 + 64 - 10 - 2 - 1 \\
 &= 123
 \end{aligned}$$

The transition value is the sum of these integrals which is  $577.5 + 218 + 123 = 918.5$  which agrees with the first transition value in the computer output.

■

## Appendix J

# Determining Minimal Transitions:2

## Two Factors per Change, Even Number of Factors

The algorithm *whichtwotochangeeven.f* finds the transition from  $\mathbf{x}_0$  to  $\mathbf{x}_1$  which has smallest value when only two controllable factors are changed at a time and the number of factors is even. When two factors are moved it is assumed that they follow a straight line path.  $L(\mathbf{x})$  represents a variance or an adjusted variance function. It is a polynomial of order 2 in  $p$  factors and can be represented by

$$L(\mathbf{x}) = A + \mathbf{x}^t \mathbf{B} + \mathbf{x}^t \mathbf{C} \mathbf{x}$$

Where  $\mathbf{C}$  is a symmetric matrix. A transition consists of the sequence of two factor changes. Its value is the sum of the line integrals on  $L(\mathbf{x})$  corresponding to the paths these factor changes take.

### **whichtwotochangeeven.f**

#### **Inputs**

The inputs are

**paths:** This is a matrix of dimensions  $n_t \times p$ . Where  $p$  is the number of controllable factors and  $n_t$  is the number of paths to be considered. A row in the path matrix represents the order that the  $p$  factors are going to be changed in e.g

a typical row for a 4 factor problem could be (2,4,1,3) which means the second and fourth factor are changed together and then the first and third.

**AA:** represents the constant,  $A$ , in the equation  $L(\mathbf{x}) = A + \mathbf{x}'\mathbf{B} + \mathbf{x}'\mathbf{C}\mathbf{x}$ . It is a scalar object.

**BB:** represents the linear terms,  $\mathbf{B}$ , in the equation  $L(\mathbf{x}) = A + \mathbf{x}'\mathbf{B} + \mathbf{x}'\mathbf{C}\mathbf{x}$ . It is a  $p \times 1$  matrix.

**CC:** represents the interaction and squared terms,  $\mathbf{C}$ , in the equation  $L(\mathbf{x}) = A + \mathbf{x}'\mathbf{B} + \mathbf{x}'\mathbf{C}\mathbf{x}$ .  $\mathbf{C}$  is a symmetric  $p \times p$  matrix where each coefficient of an interaction will appear twice and so is halved.

**start:** represents the  $p \times 1$  matrix  $\mathbf{x}_0$

**finish:** represents the  $p \times 1$  matrix  $\mathbf{x}_1$

## Outputs

As the algorithm proceeds it outputs the row number of the transition under consideration and the sum of the line integrals for the paths in that transition.

The final output is a list comprising of

- 1) a list of the row numbers of the matrix, *paths*. These row numbers correspond to rows in the *paths* matrix that are the transitions with the smallest value.
- 2) a list of the smallest transition value.

The algorithm is

```
whichtwotochangeeven.f<-
whichtwotochangeeven.f
function(paths, AA, BB, CC, start, finish)
{
  noofpaths <- dim(paths)[1]
  nooffactors <- dim(paths)[2]
  even <- T
  if((nooffactors/2) == round(nooffactors/2)) {
    even <- T
    noofits <- nooffactors
  }
  else {
    even <- F
    cat("No of factors is not even", fill = T)
```

```

    return()
  }
  minpath <- c()
  minpathlength <- 1000000000000
  currentpathlength <- 0
  for(i in 1:noofpaths) {
    currentpos <- start
    currentpathlength <- 0
    for(j in seq(1, (noofits - 1), by = 2)) {
      currentpathlength <- currentpathlength + pathintegral2(paths[i, j
        ], paths[i, (j + 1)], currentpos, AA, BB, CC, start[paths[
          i, j]], finish[paths[i, j]], start[paths[i, (j + 1)]],
          finish[paths[i, (j + 1)]])
      currentpos[paths[i, j]] <- finish[paths[i, j]]
      currentpos[paths[i, (j + 1)]] <- finish[paths[i, (j + 1)]]
    }
    cat(i, currentpathlength, fill = T)
    if(currentpathlength == minpathlength) {
      minpath <- c(minpath, i)
    }
    if(currentpathlength < minpathlength) {
      minpath <- i
      minpathlength <- currentpathlength
    }
  }
  list(paths = minpath, min.path.length = minpathlength)
}

```

### pathintegral2

This function is called by *wichtwotochange.f* and is a generic solver for line integrals of the type specified by  $L(\mathbf{x})$  when two variables are changing in a straight line path.

```

pathintegral2<-
function(x1, x2, values, AA, BB, CC, lower1, upper1, lower2, upper2)

```



```

{
  BBB <- BB
  BBB[x1, 1] <- BB[x1, 1] * (lower1 + (upper1 - lower1)/2)
  BBB[x2, 1] <- BB[x2, 1] * (lower2 + (upper2 - lower2)/2)
  CCC <- CC
  CCC[, x1] <- CC[, x1] * (lower1 + (upper1 - lower1)/2)
  CCC[x1, ] <- CC[x1, ] * (lower1 + (upper1 - lower1)/2)
  CCC[, x2] <- CC[, x2] * (lower2 + (upper2 - lower2)/2)
  CCC[x2, ] <- CC[x2, ] * (lower2 + (upper2 - lower2)/2)
  CCC[x1, x1] <- CC[x1, x1] * (lower1^2 + lower1 * (upper1 - lower1)
    + 1/3 * (upper1 - lower1)^2)
  CCC[x2, x2] <- CC[x2, x2] * (lower2^2 + lower2 * (upper2 - lower2)
    + 1/3 * (upper2 - lower2)^2)
  CCC[x1, x2] <- CC[x1, x2] * (lower2 * lower1 + (lower1 *
    (upper2 - lower2))/2 + (lower2 * (upper1 - lower1))/2 +
    ((upper1 - lower1) * (upper2 - lower2))/3)
  CCC[x2, x1] <- CCC[x1, x2]
  upppervalues <- values
  upppervalues[x1, 1] <- 1
  upppervalues[x2, 1] <- 1
  upppervalues <- matrix(upppervalues, , 1)
  ((upper1 - lower1)^2 + (upper2 - lower2)^2)^0.5 * (AA + t(upppervalues)
    %*% BBB + t(upppervalues) %*% CCC %*% upppervalues)
}

```

## Typical Input

Consider this example.

$$L(x_1, x_2, x_3, x_4) = x_1^2 + 2x_2^2 + 3x_3^2 + 4x_4^2$$

with  $\mathbf{x}_0^t = (5, 5, 5, 5)$  and  $\mathbf{x}_1^t = (-1, -1, -1, -1)$ .  $\mathbf{x}_0$  and  $\mathbf{x}_1$  are used as inputs for the variables start and finish respectively.

By looking at the equation  $L(x_1, x_2, x_3, x_4)$  it gives as inputs

$$AA = 0 \quad BB = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \quad CC = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 4 \end{pmatrix}$$

From observation the minimal transition would be to move factors 3 and 4 first, then factors 1 and 2 i.e.  $x_3$  and  $x_4$  then  $x_1$  and  $x_2$ .

The matrix below, is the input object *paths*. Each row represents a transition.

The fourth row represents the transition, move factors 2 and 3 first and then factors 1 and 4.

```

1 2 3 4
1 3 2 4
1 4 2 3
2 3 1 4
2 4 1 3
3 4 1 2

```

With  $p = 4$  factors there are  $n_p = \frac{p!}{2^2} = 6$  transitions. They are all accounted for in *paths*.

## Typical Output

The function call is *whichtwotochangeeven.f(paths, AA, BB, CC, start, finish)*.

The output is

```

1 2104.34978081116
2 1900.70302782944
3 1697.05627484771
4 1697.05627484771
5 1493.40952186599
6 1289.76276888426

```

`$paths:`

```
[1] 6
```

\$\text{min.path.length:}\$

[,1]

[1,] 1289.763

The transitions with the lowest sum of path integrals is in row 6. This is move factors 3 and 4 together then factors 1 and 2. This transition has a value of 1289.763.

### Calculations By Hand

Calculating the first transition, (1, 2, 3, 4), by hand gives

Path 1.

The first and second factors are both changed from value 5 to  $-1$  with factors 3 and 4 remaining constant at 5.

Let  $x_1 = 5 - 6t$ ,  $t = 0 \dots 1$  and  $x_2 = 5 - 6t$ ,  $t = 0 \dots 1$  so

$$\begin{aligned} & \int_0^1 (x_1^2 + 2x_2^2 + 3x_3^2 + 4x_4^2) (\dot{x}_1^2 + \dot{x}_2^2)^{\frac{1}{2}} |dt| \\ &= (6^2 + 6^2)^{\frac{1}{2}} \int_0^1 (3(5 - 6t)^2 + 75 + 100) dt \\ &= (6^2 + 6^2)^{\frac{1}{2}} \left[ 3(25t - \frac{60}{2}t^2 + \frac{36}{3}t^3) + 175t \right]_0^1 \\ &= 1663.11514935075968 \end{aligned}$$

Path 2.

The third and fourth factors are both changed from value 5 to  $-1$  with factors 1 and 2 remaining constant at  $-1$ .

Let  $x_3 = 5 - 6t$ ,  $t = 0 \dots 1$  and  $x_4 = 5 - 6t$ ,  $t = 0 \dots 1$  so

$$\begin{aligned} & \int_0^1 (x_1^2 + 2x_2^2 + 3x_3^2 + 4x_4^2) (\dot{x}_3^2 + \dot{x}_4^2)^{\frac{1}{2}} |dt| \\ &= (6^2 + 6^2)^{\frac{1}{2}} \int_0^1 (3 + 7(5 - 6t)^2) dt \\ &= (6^2 + 6^2)^{\frac{1}{2}} \left[ 3 + 7(25t - \frac{60}{2}t^2 + \frac{36}{3}t^3) \right]_0^1 \\ &= 441.234631460405632 \end{aligned}$$

The transition value is the sum of these integrals which is

$1663.1151 + 441.2346 = 2104.3498$ . This agrees with the first transition value in the computer output.

■

## Two Factors per Change, Odd Number of Factors

The algorithm *whichtwotochangeodd.f* finds the transition from  $\mathbf{x}_0$  to  $\mathbf{x}_1$  which has smallest value when two controllable factors are changed at a time. In this case there must be one single factor change. This factor change can be first, last or between any of the two factor changes. When two factors are moved it is assumed that they follow a straight line path.  $L(\mathbf{x})$  represents a variance or an adjusted variance function. It is a polynomial of order 2 in  $p$  factors and can be represented by

$$L(\mathbf{x}) = A + \mathbf{x}^t \mathbf{B} + \mathbf{x}^t \mathbf{C} \mathbf{x}$$

Where  $\mathbf{C}$  is a symmetric matrix. A transition consists of the sequence of two factor with a final one factor change if the number of controllable factors is odd. Its value is the sum of the line integrals on  $L(\mathbf{x})$  corresponding to the paths these factor changes take.

### whichtwotochangeodd.f

#### Inputs

The inputs are

**paths:** This is a matrix of dimensions  $n_t \times p$ . Where  $p$  is the number of controllable factors and  $n_t$  is the number of paths to be considered. A row in the path matrix represents the order that the  $p$  factors are going to be changed in with the last factor being the factor that changes by itself. A typical row might be ( 2, 4, 3, 5, 1) which means that factors 2 and 4 change first then factors 3 and 5. Factor 1 changes by itself. It can change first, last or between the two factor changes. In in this case the row represents 3 transitions.

**AA:** represents the constant,  $A$ , in the equation  $L(\mathbf{x}) = A + \mathbf{x}^t \mathbf{B} + \mathbf{x}^t \mathbf{C} \mathbf{x}$ . It is a scalar object.

**BB:** represents the linear terms,  $\mathbf{B}$ , in the equation  $L(\mathbf{x}) = A + \mathbf{x}^t \mathbf{B} + \mathbf{x}^t \mathbf{C} \mathbf{x}$ . It is a  $p \times 1$  matrix.

**CC:** represents the interaction and squared terms,  $\mathbf{C}$ , in the equation  $L(\mathbf{x}) = A + \mathbf{x}^t \mathbf{B} + \mathbf{x}^t \mathbf{C} \mathbf{x}$ .  $\mathbf{C}$  is a symmetric  $p \times p$  matrix where each coefficient of an interaction will appear twice and so is halved.

**start:** represents the  $p \times 1$  matrix  $\mathbf{x}_0$

**finish:** represents the  $p \times 1$  matrix  $\mathbf{x}_1$

## Outputs

As the algorithm proceeds it outputs the row number of the transition under consideration, the position of the single factor change, and the sum of the line integrals for the paths in that transition.

The final output is a list comprising two parts

1) a matrix. The first column of this matrix gives the row numbers corresponding to rows in the matrix *paths*. The second column gives the positions of where the single factor change occurs. A one means that the change occurs first, a two means it occurs before the second two factor change etc. Together they give the recommended transitions.

2) a list that gives the smallest transition value.

The algorithm is

```
whichtwotochangeodd.f
function(paths, AA, BB, CC, start, finish)
{
  noofpaths <- dim(paths)[1]
  nooffactors <- dim(paths)[2]
  even <- T
  if((nooffactors/2) == round(nooffactors/2)) {
    even <- T
    cat("Error: even number of factors", fill = T)
    return()
  }
  else {
    even <- F
    noofits <- (nooffactors - 1)
  }
  minpath <- matrix(0, 1, 2)
  minpathlength <- 1000000000000
  currentpathlength <- 0
  for(i in 1:noofpaths) {
    for(k in 1:(noofits/2 + 1)) {
      currentpos <- start
      currentpathlength <- 0
```

```

if(k == 1) {
  currentpathlength <- currentpathlength + pathintegral(
paths[i, nooffactors], currentpos, AA, BB, CC, finish[
paths[i, nooffactors]], start[paths[i, nooffactors]])
currentpos[paths[i, nooffactors]] <- finish[paths[i,
  nooffactors]]
for(j in seq(1, (noofits - 1), by = 2)) {
  currentpathlength <- currentpathlength + pathintegral2(
paths[i, j], paths[i, (j + 1)], currentpos, AA, BB,
  CC, start[paths[i, j]], finish[paths[i, j]], start[
paths[i, (j + 1)]], finish[paths[i, (j + 1)]])
currentpos[paths[i, j]] <- finish[paths[i, j]]
currentpos[paths[i, (j + 1)]] <- finish[paths[i,
  (j + 1)]]
}
cat(i, "Singleton first", currentpathlength, fill = T)
if(currentpathlength == minpathlength) {
  minpath <- rbind(minpath, matrix(c(i, 1), 1, 2))
}
if(currentpathlength < minpathlength) {
  minpath <- matrix(c(i, 1), 1, 2)
  minpathlength <- currentpathlength
}
}
if(k == (noofits/2 + 1)) {
  for(j in seq(1, (noofits - 1), by = 2)) {
    currentpathlength <- currentpathlength + pathintegral2(
paths[i, j], paths[i, (j + 1)], currentpos, AA, BB,
    CC, start[paths[i, j]], finish[paths[i, j]], start[
paths[i, (j + 1)]], finish[paths[i, (j + 1)]])
currentpos[paths[i, j]] <- finish[paths[i, j]]
currentpos[paths[i, (j + 1)]] <- finish[paths[i, (j + 1)]]
}
currentpathlength <- currentpathlength + pathintegral(

```

```

    paths[i, nooffactors], currentpos, AA, BB, CC, finish[
    paths[i, nooffactors]], start[paths[i, nooffactors]])
cat(i, "Singleton last", currentpathlength, fill = T)
if(currentpathlength == minpathlength) {
    minpath <- rbind(minpath, matrix(c(i,
    (noofits/2 + 1)), 1, 2))
}
if(currentpathlength < minpathlength) {
    minpath <- matrix(c(i, noofits/2 + 1), 1, 2)
    minpathlength <- currentpathlength
}
}
if((k != 1) & (k != (noofits/2 + 1))) {
for(j in seq(1, ((k - 1) * 2), by = 2)) {
    currentpathlength <- currentpathlength + pathintegral2(
    paths[i, j], paths[i, (j + 1)], currentpos, AA, BB,
    CC, start[paths[i, j]], finish[paths[i, j]], start[
    paths[i, (j + 1)]], finish[paths[i, (j + 1)]])
    currentpos[paths[i, j]] <- finish[paths[i, j]]
    currentpos[paths[i, (j + 1)]] <- finish[paths[i,
    (j + 1)]]
}
currentpathlength <- currentpathlength + pathintegral(
    paths[i, nooffactors], currentpos, AA, BB, CC, finish[
    paths[i, nooffactors]], start[paths[i, nooffactors]])
currentpos[paths[i, nooffactors]] <- finish[paths[i,
    nooffactors]]
for(j in seq((k * 2 - 1), (noofits - 1), by = 2)) {
    currentpathlength <- currentpathlength + pathintegral2(
    paths[i, j], paths[i, (j + 1)], currentpos, AA, BB,
    CC, start[paths[i, j]], finish[paths[i, j]], start[
    paths[i, (j + 1)]], finish[paths[i, (j + 1)]])
    currentpos[paths[i, j]] <- finish[paths[i, j]]
    currentpos[paths[i, (j + 1)]] <- finish[paths[i,

```

```

        (j + 1)]]
    }
    cat(i, "Singleton before pair ", k, currentpathlength,
        fill = T)
    if(currentpathlength == minpathlength) {
        minpath <- rbind(minpath, matrix(c(i, k), 1, 2))
    }
    if(currentpathlength < minpathlength) {
        minpath <- matrix(c(i, k), 1, 2)
        minpathlength <- currentpathlength
    }
}
}
}
list(paths = minpath, min.path.length = minpathlength)
}

```

### pathintegral2

The function *pathintegral2* is called by *whichtwotochangeodd.f*. This algorithm is also called by *whichtwotochangeeven.f* and appears earlier in this appendix.

### pathintegral

This function is called by *whichtwotochangeodd.f* and is a generic solver for line integrals of the type specified by  $L(\mathbf{x})$  when only one variable is changing. It is identical to the function of the same name in the previous appendix. It is repeated here only for completeness.

```

pathintegral<-
function(x, values, AA, BB, CC, lower, upper)
{
    BBB <- BB
    BBB[x, 1] <- BB[x, 1]/2
    CCC <- CC
    CCC[, x] <- CC[, x]/2
}

```



```

CCC[x, ] <- CCC[x, ]/2
CCC[x, x] <- CC[x, x]/3
uppervalues <- values
uppervalues[x, 1] <- upper
uppervalues <- matrix(uppervalues, , 1)
lowervalues <- values
lowervalues[x, 1] <- lower
lowervalues <- matrix(lowervalues, , 1)
abs(upper * (AA + t(uppervalues) %% BBB + t(uppervalues) %%
  CCC %% uppervalues) - lower * (AA + t(lowervalues) %% BBB
  + t(lowervalues) %% CCC %% lowervalues))
}

```

## Typical Input

This example uses the same data as the example in the previous appendix for single factor changes.

Suppose  $L(x, y, z) = 10 + x + 3y + 4z + 5x^2 + 6y^2 + 3z^2 + 4xy + 2xz + 6yz$  then  $L(x, y, z)$  can be represented by

$$L(x, y, z) = 10 + (x, y, z) \begin{pmatrix} 1 \\ 3 \\ 4 \end{pmatrix} + (x, y, z) \begin{pmatrix} 5 & 2 & 1 \\ 2 & 6 & 3 \\ 1 & 3 & 3 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix}$$

i.e.

$$A = 10, \mathbf{B} = \begin{pmatrix} 1 \\ 3 \\ 4 \end{pmatrix}, \mathbf{C} = \begin{pmatrix} 5 & 2 & 1 \\ 2 & 6 & 3 \\ 1 & 3 & 3 \end{pmatrix}$$

As input  $AA = A$ ,  $BB = \mathbf{B}$  and  $CC = \mathbf{C}$ .

The initial position is  $\mathbf{x}_0 = \begin{pmatrix} 3 \\ 2 \\ 4 \end{pmatrix}$  and input as the variable *start*.

The final position is  $\mathbf{x}_1 = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$  and is input as the variable *finish*.

The last input is the *paths* matrix. For this problem the paths in rows 1 and 4, 2 and 5, 3 and 6 are equivalent.

1	2	3
1	3	2
2	3	1
2	1	3
3	1	2
3	2	1

In row 6 the transition is to move factor 3,  $z$  and factor 2,  $y$  together and then move factor 1,  $x$ .

## Typical Output

The function call is *whichtwotochange.f(paths, AA, BB, CC, start, finish)*. The output is

```

1 Singleton first 785.741403441035
1 Singleton last 647.60771058001
2 Singleton first 668.135605693432
2 Singleton last 655.605735166021
3 Singleton first 829.888589282479
3 Singleton last 632.304934942278
4 Singleton first 785.741403441035
4 Singleton last 647.60771058001
5 Singleton first 668.135605693432
5 Singleton last 655.605735166021
6 Singleton first 829.888589282479
6 Singleton last 632.304934942278
$paths:
      [,1] [,2]
[1,]    3    2
[2,]    6    2

$min.path.length:
      [,1]
[1,] 632.3049

```

The transitions with the lowest sum of path integrals is in row 3 and 6 with the sum of 632.3. This transition is (2, 3, 1) or equivalently (3, 2, 1). Both have a corresponding 2 meaning that the single factor move occurs after the paired factor move. A one would indicate that the single factor move would happen first. In the previous appendix, where only single factor moves were considered, the same transition sequence was given as the minimum. The sum of line integrals for that transition was 807.5 which is larger.

### Calculations By Hand

Calculating, by hand, the first transition, (1, 2, 3) with the single factor change occurring last, gives

Path 1.

The first and second factors are changed from value 3 to 0 and 2 to 0 respectively with factor 3 remaining constant at 4.

Let  $x = 3t$ ,  $t = 0 \dots 1$  and  $y = 2t$ ,  $t = 0 \dots 1$  so

$$\begin{aligned}
 & \int_0^1 (10 + x + 3y + 4z + 5x^2 + 6y^2 + 3z^2 + 4xy + 2xz + 6yz) (x^2 + y^2)^{\frac{1}{2}} |dt| \\
 = & (3^2 + 2^2)^{\frac{1}{2}} \left[ 10t + \frac{3t^2}{2} + \frac{6t^2}{2} + 4 \times 4 + \frac{5 \times 9t^3}{3} + \frac{6 \times 4t^3}{3} \right. \\
 & \left. + 3 \times 4^2 + \frac{4 \times 3 \times 2t^3}{3} + \frac{2 \times 3 \times 4t^2}{2} + \frac{6 \times 2 \times 4t^2}{2} \right]_0^1 \\
 = & (3^2 + 2^2)^{\frac{1}{2}} \times \left( 10 + \frac{3}{2} + \frac{6}{2} + 16 + \frac{5 \times 9}{3} + \frac{6 \times 4}{3} \right. \\
 & \left. + 3 \times 4^2 + \frac{4 \times 3 \times 2}{3} + \frac{2 \times 3 \times 4}{2} + \frac{6 \times 2 \times 4}{2} \right) \\
 = & 524.6077
 \end{aligned}$$

Path 2.

The third factor is changed from value 4 to 1 with factors 1 and 2 both remaining constant at 0.

$$\begin{aligned}
 & \int_1^4 10 + x + 3y + 4z + 5x^2 + 6y^2 + 3z^2 + 4xy + 2xz + 6yz |dz| \\
 = & \left[ 10z + \frac{4z^2}{2} + \frac{3z^3}{3} \right]_1^4 \\
 = & 40 + 32 + 64 - 10 - 2 - 1 \\
 = & 123
 \end{aligned}$$

The transition value is the sum of these integrals which is  $524.6077 + 123 = 647.6077$ . This agrees with the second value for the first transition in the computer output.

■

# Bibliography

- [1] Albin, Susan L., Kang, Lan and Shea, Gerald, (1997). An X and EWMA Chart for Individual Observations. *Journal of Quality Technology*, **29**, 41-47.
- [2] American Supplier Institute, (1993). Achieving The Best In A Real World. *Engineering*, January 1993, p19.
- [3] American Supplier Institute, (1993). Taguchi Methods - Cutting the Cost of Quality. *Metallurgia*, February 1993, p68.
- [4] Anton, Howard, (1984). *Calculus With Analytic Geometry*, 2 ed., John Wiley & Sons.
- [5] Box, G.E.P. and Hunter, J.S., (1961). The  $2^{k-p}$  Fractional Factorials Designs Part 1, *Technometrics*, **3**, 311-351.
- [6] Box, G.E.P., Hunter, W.G., and Hunter, J.S., (1978). *Statistics for Experimenters*. John Wiley and Sons.
- [7] Box, G.E.P.,(1988). Signal-to-Noise Ratios, Performance Criteria, and Transformations (with discussion). *Technometrics*, **30**, 1-40.
- [8] Box, G.E.P. and Jones, S., (1992). Split-plot Designs For Robust Product Estimation. *Journal of Applied Statistics*, **19**, 3-26.
- [9] Box, G.E.P. and Wilson, K.B., (1951). On the Experimental Attainment of Optimum Conditions. *Journal of the Royal Statistical Society*, **12**, 1-38.
- [10] Bronson, Richard, (1973). *Modern Introductory Differential Equations*, McGraw-Hill Book Company.
- [11] Burden, Richard L. and Faires, J. Douglas, (1989). *Numerical Analysis, Fourth Edition*, PWS-Kent Publishing Company.

- [12] Davies, Owen L. (editor), (1979). *The Design and Analysis of Industrial Experiments*. Oliver and Boyd.
- [13] Engel, J, (1992). Modeling Variation in Industrial Experiments. *Applied Statistics*, **41**, 579-593.
- [14] Engel, Jan and Heule, A. Freek, (1996). A Generalized Linear Modelling Approach to Robust Design. *Technometrics*, **38**, 365-372.
- [15] Fourer, R., Gay, D.M. and Kernighan, B.W., (1993). *Ampl: A Modeling Language for Mathematical Programming*. Duxbury Press and Brooks/Cole Publishing Company.  
<http://www.ampl.com/>
- [16] Ghosh, S., and Duh, Yi-Jing, (1992). Determination of Optimum Experimental Conditions Using Dispersion Main Effects and Interactions of Factors in Replicated Factorial Experiments. *Journal of Applied Statistics*, **19**, 367-378.
- [17] Greenall, Ray, (1992). A Quality Approach Japanese Style. *Process Engineering*. June 1992, p45.
- [18] Lin, Dennis K.J. and Yu, Wanzhu, (1995). Dual Response Surface Optimization, *Journal of Quality Technology* **27**, 34-39
- [19] León, R.V., Shoemaker, A.C., and Kacker, R.N., (1987). Performance Measures Independent of Adjustment (with discussion). *Technometrics*, **29**, 253-285.
- [20] *Maple V*, (1981-1994). Release 3 for Sun SPARC, SunOS 4.13. Waterloo Software and the University of Waterloo,  
<http://www.maplesoft.com/>
- [21] McCullagh, P. and Nelder, J.A., (1989). *Generalized Linear Models*, 2 ed. *McGraw-Hill Inc.*
- [22] Michaels, S.E., (1964). The Usefulness of Experimental Design, *Applied Statistics*, **13**, 221-235.
- [23] Morton, R. Hugh, (1982). Response Surface Methodology. *The Mathematical Scientist*, **8**, 31-52.

- [24] Myers, R.H., Khuri, A.I. and Vining G., (1992). Response Surface Alternatives to the Taguchi Robust Parameter Design Approach, *The American Statistician*, **46**, 131-139.
- [25] Nair, V.N. (editor), (1992). Taguchi's Parameter Design: A Panel Discussion. *Technometrics*, **34**, 128-161.
- [26] Nelder, J.A. and Lee, Y. (1991). Generalized Linear Models For The Analysis Of Taguchi-Type Experiments, *Applied Stochastic Models And Data Analysis*, **7**, 107-120.
- [27] Pease, Bob, (1992). What's All This Taguchi Stuff, Anyhow? *Electronic Design*. June 25, 1992, 83-84.
- [28] Pease, Bob, (1992). Bob's Mailbox. *Electronic Design*. Nov 2, 1992, 71-73.
- [29] Pease, Bob, (1993). What's All This Taguchi Stuff, Anyhow? (Part II) *Electronic Design*. June 10, 1993, 85-92.
- [30] Plackett, R.L., and Burman, J.P. (1946). The Design of Optimum Multi-factorial Experiments, *Biometrika*, **33**, 303 - 325
- [31] Pledger, M.J., (1996). Observable Uncontrollable Factors in Parameter Design, *Journal of Quality Technology*, **28**, No 2, 153-162.
- [32] Pignatiello, J.J. and Ramberg J.S., (1991). Top Ten Triumphs and Tragedies of Genichi Taguchi, *Quality Engineering*, **4(2)**, 211-225.
- [33] Ross, Phillip J., (1988). *Taguchi Techniques for Quality Engineering*. McGraw-Hill Inc.
- [34] *Splus*, (1988, 1996). Version 3.4, Release 1 for Sun SPARC, SunOS 4.13. Mathsoft Inc.  
<http://www.mathsoft.com/>
- [35] Steinberg, D.M. and Bursztyn, D., (1994). Dispersion Effects in Robust-Design Experiments with Noise Factors. *Journal of Quality Technology*, **26**, 12-20.
- [36] Taguchi, Genishi and Wu, Yuiin, (1979). *Introduction to Off-Line Quality Control*, Central Japan Quality Control Association, Nagaya, Japan.

- [37] Wylie, C. Ray, Barret, Louis C., (1989). *Advanced Engineering Mathematics. Fifth Edition.* McGraw-Hill International Book Company.
- [38] Kshirsagar, Anant M, (1972) *Multivariate Analysis.* Marcell Dekker Inc, New York.
-



## Corrections

A listing of corrections to the text.

Page; Lines	Original and Corrected Text
9: 2	The Inner and Outer arrays are both level 3 Designs. The Inner and Outer arrays are both resolution 3 designs.
30: 7	Where the factors fall in this graph gives an indication of what type of factor it is. Where a factor falls in this graph gives an indication of what type of factor it is.
30: 17	The areas outside the crossarms are where both significant mean and log standard effects lie. The areas outside the crossarms are where both significant mean and log standard deviation effects lie.
99: 9	In the previous example, the conditions for $\hat{\mathbf{x}}_i^*$ to be the minimum of the variance when the mean is on target, were met. In the previous example, the conditions for $\hat{\mathbf{x}}_i^*$ to give the minimum of the variance when the mean is on target, were met.
99: 10	If $\Lambda_2$ is positive definite and $\Sigma_2$ is of rank $r$ then $\hat{\mathbf{x}}_i^{***}$ is the minimum of the MSE. If $\Lambda_2$ is positive definite and $\Sigma_2$ is of rank $r$ then $\hat{\mathbf{x}}_i^{***}$ gives the minimum of the MSE.
104: 13	And so long as $\lambda_1$ and $\lambda_2$ are chosen to be both greater then the conditions are met. And so long as $\lambda_1$ and $\lambda_2$ are chosen to be both greater than 0 then the conditions are met.
167: 21	Let $\epsilon$ have a standard Normal distribution and be independant of $z_1$ and $z_2$ and have a joint multi-variate Normal distribution. Let $\epsilon$ have a standard Normal distribution and be independant of $z_1$ and $z_2$ . Together, let $\epsilon$ , $z_1$ and $z_2$ and have a multivariate Normal distribution.

Page; Lines	Original and Corrected Text
178: 11	<p>In Figure 4.7 the Mean Square Error of <math>\hat{z}_1</math> for <math>z_1</math> has been plotted against the correlation coefficient of <math>z_1</math> and <math>z_2</math>.</p>
	<p>In Figure 4.7 the Mean Square Error of <math>\hat{z}_1</math> for <math>z_1</math> has been plotted against the correlation coefficient between <math>z_1</math> and <math>z_2</math>.</p>
179: 2	<p>The graph of the Mean Square Error for <math>\hat{z}_1</math>, the estimate of <math>z_1</math>, versus the Correlation Coefficient of <math>z_1</math> and <math>z_2</math>.</p>
	<p>The graph of the Mean Square Error for <math>\hat{z}_1</math>, the estimate of <math>z_1</math>, versus the Correlation Coefficient between <math>z_1</math> and <math>z_2</math>.</p>
178: 7	<p>In Figure 4.8 the Mean Square Error of <math>\hat{z}_2</math> for <math>z_2</math> has been plotted against the correlation coefficient of <math>z_1</math> and <math>z_2</math>.</p>
	<p>In Figure 4.8 the Mean Square Error of <math>\hat{z}_2</math> for <math>z_2</math> has been plotted against the correlation coefficient between <math>z_1</math> and <math>z_2</math>.</p>
179: 9	<p>The graph of the Mean Square Error for <math>\hat{z}_2</math>, the estimate of <math>z_2</math>, versus the Correlation Coefficient of <math>z_1</math> and <math>z_2</math>.</p>
	<p>The graph of the Mean Square Error for <math>\hat{z}_2</math>, the estimate of <math>z_2</math>, versus the Correlation Coefficient between <math>z_1</math> and <math>z_2</math>.</p>
179: 13	<p>In Figure 4.9 the Mean Square Error of <math>\hat{\epsilon}</math> for <math>\epsilon</math> has been plotted against the correlation coefficient of <math>z_1</math> and <math>z_2</math>.</p>
	<p>In Figure 4.9 the Mean Square Error of <math>\hat{\epsilon}</math> for <math>\epsilon</math> has been plotted against the correlation coefficient between <math>z_1</math> and <math>z_2</math>.</p>
179: 2	<p>The graph of the Mean Square Error for <math>\hat{\epsilon}</math>, the estimate of <math>\epsilon</math>, versus the Correlation Coefficient of <math>z_1</math> and <math>z_2</math>.</p>
	<p>The graph of the Mean Square Error for <math>\hat{\epsilon}</math>, the estimate of <math>\epsilon</math>, versus the Correlation Coefficient between <math>z_1</math> and <math>z_2</math>.</p>
180: 9	<p>2) when there has been a change in the system e.g. a change in process mean or variance.</p>
	<p>2) when there has been a change in the system e.g. a change in process mean or an increase in variance.</p>