

# **Complexity Reduction in High-Dimensional Multi-Objective Optimisation**

Thesis submitted for candidature for the degree of Ph.D.

Robert John Lygoe

May, 2010

Department of Automatic Control & Systems Engineering,  
The University of Sheffield.



# Summary

A novel process has been developed for reducing complexity in real-world, high-dimensional, multi-objective optimisation problems. This approach relies on being able to identify and exploit local harmony between objectives to reduce dimensionality. To achieve this, a systematic and modular process has been designed to cluster the Pareto-optimal front and apply a rule-based Principal Component Analysis including preference articulation for potential objective reduction.

Proof-of-principle is demonstrated on a simplified, real-world, automotive diesel engine calibration optimisation problem comprising three objectives. Some objective reduction was achieved in one cluster, from which a selected solution performed better when compared with the parent three-objective problem.

On a six-objective version of the diesel problem, the complexity reduction process resulted in three and four objective sub-problems. In the former, a significant improvement was achieved in one of the retained objectives at very little cost to the others.

A further case study comprised a ten-objective gasoline engine cold start calibration optimisation problem, including sensitivity objectives related to a control system actuator, which exhibited significant variation. For brevity, efficiency and to support future software development, a mathematical notation was developed for the clustering and objective reduction analysis. To address the computational demands, a parallel computing cluster was utilised and a parallel island-based optimisation algorithm was developed. The complexity reduction process consists of four stages and progressively reduces objective dimensionality where evidence of local objective harmony exists. It involves the calibration engineer at various stages to advise on objective priorities and to discard clusters containing solutions of no interest. This process culminated in two sub-problems, one of three and one of four conflicting objectives. A comparison of the resulting Pareto-optimal front, selected preferred solution and an independently generated, manually tuned calibration was made for each of the two sub-problems. In both cases, the preferred solution outperformed the independent calibration.

# Acknowledgements

I would like to thank my academic supervisor, Prof. Peter Fleming, for providing expert strategic direction and my industrial supervisor, Dr. Mark Cary, for invaluable technical guidance and support. I am very grateful to Ford Motor Company for funding this Ph.D. and for providing management support. This work would not have been possible without the support of various colleagues at Ford, notably Dr. Andrew Emtage, Martin England, Mark Skilling and Jeff Smith. My thanks also to Sham Ahmed, Paul Kerr-Delworth, Matt McDonnell and Ian Noell at The MathWorks for providing the Matlab<sup>®</sup> software used in this research. I would also like to thank Isabel Cary for all her proof-reading and the very generous and hospitable use of the study as well as Ian and Stella Ross for the use of their front room to get this finished.

Lastly, but most importantly, I am forever indebted to my family for their enduring love and in particular, to Jan, who has put up with me doing this over several years and without whose support, this thesis would not have been possible.

# Statement of Originality

Unless otherwise stated in the text, the work described in this thesis was carried out solely by the candidate. None of this work has already been accepted for any other degree, nor is it being concurrently submitted in candidature for any degree.

Candidate: \_\_\_\_\_

Robert J. Lygoe

Supervisor: \_\_\_\_\_

Peter J. Fleming

*This thesis is dedicated to Rachel, Amy and Eddie. It's surprising what you can achieve when you put your mind to it.*

# Nomenclature

## Acronyms

<b>CAO</b>	Convergence Accelerator Operator
<b>CFND</b>	Constrained Fast Non-Dominated
<b>CIDI</b>	Compression Ignition Direct Injection
<b>CO</b>	Carbon Monoxide
<b>CO<sub>2</sub></b>	Carbon Dioxide
<b>CPU</b>	Central Processing Unit
<b>CR</b>	Cluster verification Rule
<b>DM</b>	Decision Maker
<b>EA</b>	Evolutionary Algorithm
<b>EGR</b>	Exhaust Gas Recirculation
<b>EMO</b>	Evolutionary Many-Objective Optimisation
<b>FIE</b>	Fuel Injection Equipment
<b>GDI</b>	Gasoline Direct Injection
<b>HC</b>	Hydrocarbon
<b>ICA</b>	Independent Component Analysis
<b>MCDM</b>	Multi-Criteria Decision Making
<b>MOODM</b>	Many-Objective Optimisation Decision-Making
<b>MOEA</b>	Multi-Objective Evolutionary Algorithm
<b>MOO</b>	Multi-Objective Optimisation
<b>NO<sub>x</sub></b>	Oxides of Nitrogen
<b>NMEP</b>	Net Mean Effective Pressure

<b>OR</b>	Objective reduction Rule
<b>Parts</b>	Particulates
<b>PC</b>	Principal Component
<b>PCA</b>	Principal Component Analysis
<b>PFI</b>	Port Fuel Injection
<b>pMOEA</b>	parallel Multi-Objective Evolutionary Algorithm
<b>POP</b>	Pareto-Optimal Population
<b>PPA</b>	Progressive Preference Articulation
<b>PPA<sub>FF</sub></b>	Progressive Preference Articulation method of Fonseca and Fleming (1998a)
<b>RBF</b>	Radial Basis Function
<b>RMS</b>	Root Mean Square
<b>ROI</b>	Region Of Interest
<b>RPM</b>	Revolutions Per Minute
<b>SFC</b>	Specific Fuel Consumption
<b>SLH</b>	Stratified Latin Hypercube
<b>SN</b>	Signal-to-Noise
<b>SVD</b>	Singular Value Decomposition

## Empirical Models and Inputs

<b>cyc_612_RMS (612R)</b>	Combustion variation metric for cycles 6-12 (bar)
<b>F_model (F)</b>	Fuel quantity (unitless)
<b>neg_25_INT (25I)</b>	Negative run-up combustion intensity for cycles 2-5 (bar)



<b>neg_612_INT (612I)</b>	Negative run-up combustion intensity for cycles 6-12 (bar)
<b>Peak_Flare_Speed (PFS)</b>	Peak Flare Speed (rpm)
<b>sf_cyc_25_RMS_dFP_abs (s25R)</b>	Absolute value of sensitivity of combustion variation metric for cycles 2-5 to fuel pressure (bar/MPa)
<b>sf_cyc_612_RMS_dFP_abs (s612R)</b>	Absolute value of sensitivity of combustion variation metric for cycles 6-12 to fuel pressure (bar/MPa)
<b>sf_cyc_25_INT_dFP_abs (s25I)</b>	Absolute value of sensitivity of combustion intensity metric for cycles 2-5 to fuel pressure (bar/MPa)
<b>sf_cyc_612_INT_dFP_abs (s612I)</b>	Absolute value of sensitivity of combustion intensity metric for cycles 6-12 to fuel pressure (bar/MPa)
<b>AIR</b>	Inducted air mass flow (kg/h)
<b>DEC</b>	Exponential decay in injected fuel quantity (unitless)
<b>EOI</b>	End Of Injection crankshaft angle timing (degrees BTDC)
<b>F</b>	Injected fuel quantity (unitless)
<b>FP</b>	Fuel Pressure (MPa)
<b>SPK2</b>	Crankshaft angle timing of ignition (degrees BTDC)

## Symbols

$\beta$	Threshold of engineering significance for $\Omega_k$
$\chi_{df}^2$	Chi-squared distribution with $df$ degrees of freedom
$\Delta x_{inp}$	Small perturbation to the <i>inpth</i> objective function input, $x$

${}_p\Delta_-$	The set of objectives retained for further comparison with negative eigenvector coefficients for the $p$ th PC
${}_p\Delta_+$	The set of objectives retained for further comparison with positive eigenvector coefficients for the $p$ th PC
$\gamma$	Sub-sample size
$\lambda_p$	Eigenvalue associated with $p$ th PC
$\Omega_k$	The largest absolute normalised difference in centres between the reference and the $k$ th sub-sampled cluster
$\Omega_k(\psi)$	Denotes the explicit value of $\Omega_k$
$\Phi_k$	The maximum absolute difference in correlations for the $k$ th sub-sample
$\Phi_k(\omega)$	Denotes the explicit value of $\Phi_k$
$\tau$	Convergence tolerance of clustering algorithm
$\theta_p$	The critical value for the magnitude of an eigenvector coefficient for the $p$ th PC
$\varphi$	Maximum number of iterations of clustering algorithm
$\xi_j$	Number of final clusters for $j$ th clustering run
$\zeta$	Set of objective indices
$\exists$	There exists
$\forall$	For all
$\in$	Is a member of
$\vee$	Logical OR
$\wedge$	Logical AND
$A$	Eigenvectors of correlation matrix
${}_pA_-$	The set of eigenvector coefficients of negative sign, whose magnitude exceeds $\theta_p$
${}_pA_+$	The set of eigenvector coefficients of positive sign, whose magnitude exceeds $\theta_p$

$C_c$	Correlation matrix of cluster, $c$
$C_d$	Correlation matrix of data, $d$
$C_k$	Matrix of unique elements of the correlation matrices for the $k$ th sub-sample
$C_p$	Correlation matrix of population, $p$
$ce$	Combustion Event
$ci$	Initial number of Clusters
$cf$	Final number of Clusters
${}_kD_i$	The Pareto-optimal solutions for the $k$ th sub-sample and the $i$ th cluster
$df$	Degrees of Freedom
$d_w$	NMEP deviation for an individual combustion event
$DN$	Instantaneous combustion deviations (bar)
$E$	The set of objective indices excluded (or discarded) in future analyses
$F$	Subset of $E$ excluding misfire or other aberrant combustion data
$ft$	Savitzky-Golay filter order
$g$	Preference vector
$H$	The set of observed combustion events
$h$	Proportion of data retained by FAST-MCD
$I$	The set of objective indices included (or retained) in future analyses
$i$	Cluster Priority
$inp$	Objective function input
$J$	Combustion intensity (bar)
$k$	Sub-sample
$l_m$	Mahalanobis distance
$lo_m$	Lower limit on engineering significance for the $m$ th objective
$l_p$	$p$ th eigenvalue

$M_k$	Matrix of cluster centres for the $k$ th sub-sample
$M_c$	Centres of cluster, $c$
$M_d$	Centres of data, $d$
$m$	Objective
$N_k$	Matrix of reciprocal cluster centres for the $k$ th sub-sample
$\mathbb{N}$	The set of natural numbers
$nc$	Number of Clusters
$nm$	NMEP threshold parameter
$nn$	Nearest Neighbours
$n_l$	Number of samples to the left of the $x_{ce}$ data
$nobj$	Number of objectives
$np$	Number of Processors
$npc$	Number of Principal Components
$n_r$	Number of samples to the right of the $x_{ce}$ data
$nv$	Number of Variables
$p$	Principal Component
$R$	The vector of objective priorities
$\mathbb{R}$	The set of real numbers
${}_kR_i$	The correlation matrix for the $k$ th sub-sample and the $i$ th cluster
$R_a^b$	RMS of instantaneous NMEP values over cycles $[a, b]$ (bar
$r$	Priority
$r_i$	The priority of the $i$ th objective
$rpc$	Number of Retained Principal Components
$S$	Sample covariance
$s$	The first event for which normal combustion occurs
$tc$	Threshold for Cumulative percentage of variation

$tp$	Threshold for Proportion of variation
$\mathbf{u} \prec_{\mathbf{g}} \mathbf{v}$	vector $\mathbf{u}$ is preferable to vector $\mathbf{v}$ given a preference vector $\mathbf{g}$
$\mathbf{u} \equiv_{\mathbf{g}} \mathbf{v}$	vector $\mathbf{u}$ is equivalent to vector $\mathbf{v}$ given a preference vector $\mathbf{g}$
$\mathbf{u}_{(}$	The components of objective vector $\mathbf{u}$ which meet their corresponding goals
$\mathbf{u}_{(}$	The components of objective vector $\mathbf{u}$ which violate their corresponding goals
$up_m$	Upper limit on engineering significance for the $m$ th objective
$V_k^{nobj}$	A clustering analysis for $nobj$ objectives and the $k$ th sub-sample
$\mathbf{v}^p$	Eigenvector associated with $p$ th PC
$\mathbf{v}^{p'}$	Rotated eigenvector associated with $p$ th PC
$W$	The set of NMEP values for all combustion events from first fire onwards
$x_c$	Cluster centre
$x_{ce}$	NMEP for the $ce$ th combustion event (bar)
$x_f$	Individual NMEP member of the set F (bar)
${}_kx_{\gamma m}$	Pareto-optimal solutions for the $k$ th cluster, $\gamma$ sub-sample size and $m$ th objective
$x_i$	Cluster data
$x_w$	Individual NMEP member of the set W (bar)
$y_{ce}$	Savitzky-Golay filtered $x_{ce}$ (bar)
$z$	Vector of Principal Components



# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Motivation . . . . .	1
1.2	Research Objectives . . . . .	2
1.3	Outline of the Thesis . . . . .	3
1.4	Contributions . . . . .	4
<b>2</b>	<b>Literature Review including Case Study Background</b>	<b>7</b>
2.1	Introduction to Many-Objective Optimisation . . . . .	7
2.2	Issues in Many-Objective Optimisation . . . . .	10
2.3	Search Efficiency . . . . .	11
2.3.1	Methods for increasing selective pressure . . . . .	12
2.3.2	Different Fitness Functions. . . . .	14
2.3.3	The Convergence Acceleration Operator (CAO). . . . .	15
2.3.4	Parallel MOEAs. . . . .	16
2.4	Visualisation . . . . .	19
2.4.1	The Scatter-Plot . . . . .	20
2.4.2	The Parallel Coordinates Plot . . . . .	21
2.4.3	Recent Visualisation Approaches . . . . .	22
2.5	Pertinency . . . . .	23
2.5.1	<i>A priori</i> Preference Articulation . . . . .	23
2.5.2	<i>A posteriori</i> Preference Articulation . . . . .	24
2.5.3	Progressive Preference Articulation . . . . .	25
2.6	Harmony, Conflict and Independence . . . . .	26
2.6.1	Definitions . . . . .	26
2.6.2	Global and Local Harmony and Conflict . . . . .	27

2.7	Dimension Reduction . . . . .	28
2.7.1	Linear Methods . . . . .	29
2.7.2	Non-Linear Methods . . . . .	31
2.8	The Research Gap . . . . .	32
2.9	Review of Relevant Automotive Powertrain Technologies . . . . .	35
2.9.1	Comparison of Modern Diesel to Spark Ignition Engines . . . . .	36
2.9.2	Gasoline Direct Injection Engine Technology . . . . .	38
2.9.3	Engine Calibration . . . . .	40
2.9.4	The Need for Model-Based Approaches . . . . .	41
<b>3</b>	<b>Objective Reduction Methods</b>	<b>43</b>
3.1	Introduction . . . . .	43
3.2	Partitioning of the Pareto-Optimal Front using Clustering . . . . .	43
3.3	Clustering . . . . .	44
3.3.1	Components of a Clustering Task . . . . .	45
3.3.2	Requirements of a Clustering Algorithm . . . . .	46
3.3.3	A Review of Clustering Approaches . . . . .	47
3.3.4	The k*-Means Clustering Algorithm . . . . .	49
3.3.5	k*-Means Simulation Testing . . . . .	50
3.3.6	Robust Clustering . . . . .	50
3.4	Exploiting Linear Associations for Dimension Reduction using PCA . . . . .	53
3.5	Generation of PCA . . . . .	53
3.5.1	PCA Definition . . . . .	53
3.5.2	PCA Properties . . . . .	54
3.5.3	Rotation of Principal Components . . . . .	59
3.5.4	A Graphical Explanation of PCA . . . . .	59
3.5.5	Rules for Variable Reduction . . . . .	63
3.5.6	Hyper-ellipsoidal Constraint on Principal Components . . . . .	65
3.6	Summary . . . . .	66
<b>4</b>	<b>The Many-Objective Optimisation Decision-Making Process</b>	<b>67</b>
4.1	Introduction . . . . .	67
4.2	The MOODM Process Description . . . . .	69
4.3	Generate Pareto-Optimal Population . . . . .	69



4.3.1	Apply Objective Goals and Priorities . . . . .	70
4.3.2	The Mathematical Definition of the Comparison Operator . . . .	71
4.3.3	The Migration of Preferability to NSGAII . . . . .	74
4.4	Cluster Population . . . . .	77
4.4.1	k*-Means Algorithm Parameters . . . . .	77
4.4.2	Stochastic Aspects . . . . .	77
4.4.3	Cluster Verification Rules . . . . .	78
4.5	Apply PCA to identify Redundant Objectives within each Cluster . . .	79
4.6	Continue Optimisation per Cluster with Reduced Objectives . . . . .	80
4.7	A Simplified Real-World Example . . . . .	81
4.7.1	Problem Formulation . . . . .	81
4.7.2	Three-Objective Diesel Problem Results . . . . .	83
4.8	Summary . . . . .	95
<b>5</b>	<b>Case Study: Diesel Engine Base Calibration Optimisation</b>	<b>97</b>
5.1	Introduction . . . . .	97
5.2	Six Objective Diesel Problem Formulation . . . . .	98
5.3	Six Objective Diesel Problem Results . . . . .	99
5.3.1	Clustering and Verification . . . . .	99
5.3.2	Principal Component Analysis and potential Objective Reduction	102
5.3.3	Conclusions from the Objective Reduction Process . . . . .	110
5.4	Objective Reduction applied without Clustering . . . . .	112
5.5	The Cut-off Threshold for selecting Principal Components . . . . .	116
5.6	An Application of Progressive Preference Articulation . . . . .	118
5.7	Summary . . . . .	123
<b>6</b>	<b>Case Study: Gasoline Engine Cold Start Calibration Optimisation</b>	<b>127</b>
6.1	Introduction . . . . .	127
6.1.1	Cold Start Profile Descriptions . . . . .	130
6.2	A Review of Sensitivity Analysis Approaches . . . . .	132
6.2.1	Statistical Sensitivity Analysis Approaches . . . . .	132
6.2.2	Mathematical Sensitivity Analysis Approaches . . . . .	133
6.3	Definition of Sensitivity Objective Functions . . . . .	135
6.4	pMOEA Implementation . . . . .	136

6.5	Problem Formulation . . . . .	139
6.5.1	Combustion Quality Metrics . . . . .	140
6.5.2	Optimisation Problem . . . . .	142
6.6	Results from the Objection Reduction Process - 1st Stage . . . . .	145
6.6.1	Clustering and Verification . . . . .	147
6.6.2	PCA and Potential Objective Reduction . . . . .	148
6.7	Results from the Objection Reduction Process - 2nd Stage . . . . .	151
6.7.1	Optimisation with Reduced Objectives - 2nd Stage . . . . .	151
6.7.2	Clustering and Verification with Reduced Objectives - 2nd Stage	153
6.7.3	PCA and Potential Objective Reduction - 2nd Stage . . . . .	154
6.8	Results from Objection Reduction Process - 3rd Stage . . . . .	155
6.8.1	Further Optimisation with Reduced Objectives - 3rd Stage . . .	155
6.8.2	Clustering and Verification with Reduced Objectives - 3rd Stage	155
6.8.3	PCA and Potential Objective Reduction - 3rd Stage . . . . .	157
6.9	Results from the Objection Reduction Process - 4th Stage . . . . .	161
6.9.1	Further Optimisation with Reduced Objectives - 4th Stage . . .	161
6.9.2	Clustering and Verification with Reduced Objectives - 4th Stage	161
6.9.3	PCA and Potential Objective Reduction - 4th Stage . . . . .	161
6.10	Conclusions from the Objective Reduction Process . . . . .	163
6.10.1	Selection of Final Solutions . . . . .	166
6.11	Summary . . . . .	173
<b>7</b>	<b>Conclusions and Future Work</b>	<b>175</b>
7.1	Conclusions . . . . .	175
7.2	Future Work . . . . .	180
<b>A</b>	<b>Details of k*-Means Simulation Testing</b>	<b>183</b>
<b>B</b>	<b>The Comparison Operator</b>	<b>191</b>
B.1	Some Examples explaining how Different Parts of the Comparison Op- erator work . . . . .	191
B.2	Multi-Objective Decision Making Strategies . . . . .	195
B.2.1	Pareto . . . . .	195
B.2.2	Lexicographic . . . . .	195

B.2.3	Constrained Optimisation . . . . .	197
B.2.4	Constraint Satisfaction . . . . .	198
B.2.5	Goal Programming(1) . . . . .	200
B.2.6	Goal Programming(2) . . . . .	202
<b>C</b>	<b>An Application of Savitzky-Golay Smoothing to Combustion Data</b>	<b>205</b>
<b>D</b>	<b>A Mathematical Notation for the Clustering Verification Rules</b>	<b>209</b>
D.1	A Compact Notation for Cluster Analyses used in CR1 and CR2 . . . . .	209
D.2	Comparison of Cluster Centres and Correlations in CR3 . . . . .	210
D.2.1	The Consistent Cluster Data Membership Condition . . . . .	210
D.2.2	Introduction of <i>vec</i> and <i>vech</i> Operators. . . . .	211
D.2.3	Application of <i>vec</i> and <i>vech</i> Operators to Cluster Comparisons. . . . .	211
D.3	Cluster Bound Notation used in CR4 . . . . .	213
D.4	An Example Application of the Clustering Notation . . . . .	214
<b>E</b>	<b>A Mathematical Notation for the Objective Reduction Rules</b>	<b>217</b>
E.1	The Included and Excluded Objective Sets . . . . .	217
E.2	The Sign and Magnitude of the PC Eigenvector Coefficients . . . . .	218
E.3	The Relative Magnitude of the PC Eigenvector Coefficients . . . . .	218
E.4	Denoting Objective Priorities . . . . .	220
E.5	Composite Statements . . . . .	220
<b>F</b>	<b>Detailed Results from Objective Reduction Process - 1st Stage</b>	<b>223</b>
F.1	Clustering and Verification . . . . .	223
F.1.1	Establish Reference Clusters . . . . .	223
F.1.2	Establish Sub-sampled POP Size . . . . .	224
F.1.3	Check Cluster Bounds . . . . .	229
F.2	PCA and Potential Objective Reduction . . . . .	230
F.2.1	Retaining PCs to account for 95% of the Total Variation . . . . .	230
F.2.2	Retaining PCs to account for 90% of the Total Variation . . . . .	234
F.2.3	Retaining PCs to account for 86% of the Total Variation . . . . .	236
<b>G</b>	<b>Detailed Results from Objective Reduction Process - 2nd Stage</b>	<b>239</b>
G.1	Clustering and Verification with Reduced Objectives - 2nd Stage . . . . .	239

G.2	PCA and Potential Objective Reduction - 2nd Stage . . . . .	244
G.2.1	Retaining PCs to account for 99% of the Total Variation . . . . .	245
G.2.2	Retaining PCs to account for 95% of the Total Variation . . . . .	247
G.2.3	Retaining PCs to account for 85% of the Total Variation . . . . .	249
<b>H</b>	<b>Detailed Results from Objective Reduction Process - 3rd Stage</b>	<b>253</b>
H.1	PCA and Potential Objective Reduction - 3rd Stage . . . . .	253
H.1.1	Retaining PCs to account for 99% of the Total Variation . . . . .	253
H.1.2	Retaining PCs to account for 95% of the Total Variation . . . . .	257
H.1.3	Retaining PCs to account for 85% of the Total Variation . . . . .	259
<b>I</b>	<b>Detailed Results from Objective Reduction Process - 4th Stage</b>	<b>261</b>
I.1	PCA and Potential Objective Reduction - 4th Stage . . . . .	261
I.1.1	Retaining PCs to account for 95% of the Total Variation . . . . .	261
I.1.2	Retaining PCs to account for 85% of the Total Variation . . . . .	263
<b>References</b>		<b>264</b>

# Chapter 1

## Introduction

### 1.1 Motivation

During the process of carrying out automotive engine calibration, it is common to come across trade-off problems, that is, optimisation problems comprising two or more competing objectives. In the automotive market there is ever-increasing customer demand for more fuel efficient, higher performance, increased refinement and reliability at low cost. These requirements combined with evermore stringent exhaust emissions legislation and fierce competition amongst automotive manufacturers has led to more complex engine technologies. This has, in turn, driven the development of correspondingly complex control systems, with more actuator variables and more engine responses or objectives to be optimised and traded-off.

Historically, optimisations were formulated as single objective problems, which were solved using methods available at the time, *e.g.* gradient-based or direct search algorithms. These approaches have a number of weaknesses including a tendency to get stuck in local optima, the fact that they are often designed to be problem-specific and they require multiple runs to generate a family of solutions as required for a multi-objective optimisation by definition (Deb, 2001). By contrast, Evolutionary Algorithms (EAs) evolve a population of solutions to search for the optimal trade-off or Pareto-optimal front. Such methods are able to produce a diverse set of solutions in one run of the optimiser and are well suited to multi-objective problems.

EAs have been mostly applied to two- or three-objective optimisations, the results of which are straightforward to visualise in low dimensions. However, real-world

and in particular, modern, complex engine calibration optimisation problems can involve significantly more than two or three objectives, termed *many-objective optimisations*. For such high-dimensional problems, multi-objective EAs have issues with lack of effective search, potentially large population size required (may be computationally expensive) and visualisation of the solutions, which may be sparse (Fleming *et al.*, 2005).

Research to address these issues has primarily concentrated on algorithmic developments to improve the search effectiveness. For example, repeated single-objective EA optimisations have been shown to be effective in approximating the Pareto front (Hughes, 2005). Nevertheless, even if a Pareto-optimal population has been generated, the subsequent decision-making process to select a preferred solution has received little attention, particularly in the case of many-objective problems.

This thesis addresses this research gap by proposing a practical Many-Objective Optimisation Decision-Making (MOODM) process. This is applied to real-world problems comprising automotive engine calibration optimisations. A relatively low problem (objective) dimension is considered initially to demonstrate the principle and its effectiveness. Subsequently, the proposed process is applied to two further problems involving six and ten objectives. In the latter case study, several enhancements are introduced to improve efficiency, generate optimal and robust solutions and provide more flexibility in the objective reduction process. With all the problems, final, preferred, optimal solutions are selected and compared.

## 1.2 Research Objectives

The aims of this research are as follows:

- Apply multi-objective evolutionary algorithms to many-objective real-world problems, few examples of which have been published.
- Develop and apply multi-objective decision-making processes to evolutionary optimisation problems to ultimately select a single preferred solution.
- Develop and apply an intuitive results presentation to assist the decision-maker in many-objective problems.

## 1.3 Outline of the Thesis

Chapter 2 provides a review of the literature relevant to this thesis. After an introduction to many-objective optimisation, current issues are described and remedial measures discussed. Subsequently, a practical MOODM process is proposed and used to compare current remedial measures and suggest promising approaches. An introduction to some current automotive powertrain technologies follows, together with a description of engine calibration and model-based approaches.

Chapter 3 describes how local harmony is identified and exploited for potential objective reduction. It introduces the main elements of clustering to partition the Pareto-optimal front and a Principal Component Analysis (PCA) based approach to determine objective dependencies.

Chapter 4 details the proposed MOODM process comprising optimisation, clustering and objective reduction elements. An explanation is provided of how the widely-used multi-objective evolutionary algorithm, NSGAII (Deb, Pratap, Agarwal and Meyarivan, 2002) was modified to incorporate the Progressive Preference Articulation method of Fonseca and Fleming (1998a). Subsequently, the proposed dimension reduction process is demonstrated on a simplified, real-world engine calibration optimisation problem.

The first of two calibration optimisation case studies is provided in Chapter 5. This comprises a six-objective optimisation of a diesel engine calibration. The proposed objective reduction process is applied in detail including preference articulation to direct the search and allow the most important objectives to be retained. Several further studies are conducted to explore: applying objective reduction to the whole population rather than locally, varying the threshold for retaining Principal Components (PCs) and applying progressive preference articulation.

Chapter 6 details the second calibration optimisation case study. This comprises a ten-objective gasoline engine cold start transient calibration optimisation to which the proposed MOODM process is applied. Four significant process enhancements are introduced:

- Parallel computing to address the computational demands of a large population and batch processing of clustering runs.
- A mathematical notation for the clustering verification and objective reduction

rules for clarity, brevity and efficiency.

- Sensitivity objectives to allow simultaneous search for optimal and robust solutions.
- Variation of the threshold for retaining PCs to provide flexibility in reducing objectives.

These improvements are implemented and applied in a multi-stage optimisation in conjunction with an experienced calibration engineer to progress towards smaller sub-problems from which preferred solutions are selected and compared.

Conclusions based on the thesis chapter structure are described in Chapter 7 together with suggestions for future research.

## 1.4 Contributions

The main contributions of this thesis are:

1. **A novel, systematic and modular dimension reduction process.** This comprises search, clustering and PCA-based objective reduction elements. These elements are specified in this thesis, but the process has been designed so that any one of them can be replaced with a suitable alternative method. This process, combined with visualisation approaches, assisted the decision-maker in selecting solutions, which outperformed those resulting from a hand-tuning process.
2. **Novel exploitation of local objective harmony for dimension reduction.** Partitioning of the Pareto-optimal front into clusters in which a PCA-based analysis is used to identify and, if possible, exploit objective harmony to reduce objective dimension.
  - (a) **Definition of clustering verification rules.** These have been designed to verify the correct number and location of clusters in the Pareto-optimal front. They include a sub-sampling approach to identify the smallest, most computationally efficient population suitable for clustering.
  - (b) **Definition of objective reduction rules.** These have been developed to objectively and systematically identify and retain the most influential objectives within each retained Principal Component.



- (c) **A new mathematical notation for the clustering and objective reduction rules.** These have been introduced for clarity, brevity and efficiency in a multi-stage optimisation, which arises from high-dimensional problems. In addition, such notation lends itself to future software development to minimise human error and enable process automation.

3. **Inclusion of sensitivity objectives in the optimisation.** The definition and inclusion of such objectives allow simultaneous and efficient search for solutions providing optimal trade-offs between maximising performance and minimising sensitivity to background noise.

Further minor contributions that resulted from the research work in this thesis are:

- i. **Objective priority use in objective reduction.** Objective priorities provided by the calibration engineer were used to discriminate between equivalent objectives selected using the PCA-based objective reduction rules, so that the highest priority, most influential objectives were retained.
- ii. **Flexibility in reducing the number of objectives.** Varying the threshold used for selecting Principal Components may affect the number of objectives retained using the objective reduction rules. This can provide flexibility in the dimension reduction process.
- iii. **Inclusion of Pertinency in an MOEA.** NSGAII has been modified to incorporate the progressive preference articulation method of Fonseca and Fleming (1998a). This allows the decision maker to zoom in to the region of interest by intuitively specifying objective priorities and goals.
- iv. **New parallel MOEA.** A new island-based parallel version of the modified NSGAII has been developed for efficient evaluation of large populations, which is one approach to overcome the search effectiveness issues of the serial NSGAII.
- v. **Parallel computing applied to clustering.** The clustering process is randomly initialised and thus needs to be run several times from different values for the initial number of clusters. This can be computationally demanding for large populations

used in this thesis for high-dimensional problems. A batch processing approach exploiting a distributed computing network is essential to make this task practical for high-dimensional problems, such as the ten-objective cold start optimisation.

- vi. **Use of a cluster boundary constraint.** The resulting centres and covariance data resulting from the clustering process have been used to define a hyper-ellipsoidal constraint on a cluster in an optimisation in an attempt to preserve objective correlations within the cluster. This is necessary, as it is important to verify that in subsequent optimisations, any discarded objectives do not deteriorate.
- vii. **Definition of dynamic measures of combustion quality.** These comprised smoothed, dynamic measures of combustion intensity and variation. Respectively, these are equivalent to maximising the energy available to accelerate the engine during the ‘run-up’ and minimising the combustion variation to give a smooth engine response.

## Chapter 2

# Literature Review including Case Study Background

The purpose of this chapter is to review the relevant technical literature to establish the ‘state-of-the-art’ and in particular, to identify any existing weaknesses. This will provide a reference framework against which the value of the author’s research can be assessed.

After a brief introduction to engineering optimisation approaches in Section 2.1, there follows in Section 2.2, a discussion of current issues in high-dimensional or many-objective optimisation succeeded by sections on corresponding remedial measures. A *practical many-objective optimisation decision-making process* (from hereon defined as a practical MOODM process) is then proposed in Section 2.8, against which the current countermeasures are reviewed and from which promising approaches suggested. The chapter concludes with an overview in Section 2.9 of some of the current automotive powertrain technologies, how they lead to multi-objective optimisation problems and how these may be solved efficiently utilising a model-based approach.

### 2.1 Introduction to Many-Objective Optimisation

Historically, engineering optimisation problems were formulated as single objective problems and relied on the use of gradient-based or direct search methods (Deb, 1995), extensive reviews of which can be found in Rao (1996); Ravindran *et al.* (2006). Such methods have a number of weaknesses including:

- The choice of starting position may have a strong influence on convergence to an optimal solution; *i.e.* a poor choice may result in the optimiser getting stuck at local optima, which often exist in engineering problems (Deb, 1999). This issue can be addressed by starting the optimiser from a number of different initial positions, but this is inefficient and provides no guarantee of that the global optima will be found.
- They are often designed to be problem specific.
- They cannot always realise the efficiencies offered by parallel computation.
- They are not efficient on problems with discrete variables.

Real-world engineering problems often involve the simultaneous solution of several competing objectives subject to constraints (*e.g.* Fleming *et al.* (2005)). For problems with multiple conflicting objectives, there is no one single optimal solution. Instead, there is a family of solutions, where each solution represents a compromise, or trade-off, between the competing objectives, and which cannot be improved upon with respect to all objectives. To distinguish these solutions from inferior ones, many multi-objective optimisation algorithms make use of the principle of *Pareto-dominance*. One definition (Deb, 2001) is that given two solutions  $x^1$  and  $x^2$ ,  $x^1$  dominates  $x^2$  if  $x^1$  is no worse than  $x^2$  in all  $nobj$  objectives and strictly better than  $x^2$  in at least one objective:

**Definition 2.1 (Pareto-dominance)** *A solution  $x^1$  dominates another solution  $x^2$ , (denoted by  $x^1 \prec x^2$ ) if and only if,*

$$\forall i \in \{1, \dots, nobj\}, \quad x_i^1 \leq x_i^2 \quad \wedge \quad \exists i \in \{1, \dots, nobj\} : x_i^1 < x_i^2.$$

Furthermore, if there is no feasible solution  $x^2$  which dominates  $x^1$ , then  $x^1$  is a *Pareto-optimal* or *non-dominated* solution. This is illustrated graphically in Figure 2.1. The objective vectors corresponding to the set of all Pareto-optimal solutions is known as the *Pareto-optimal front* or *trade-off surface*.

Three requirements of solutions from a multi-objective optimisation (MOO) algorithm are (Purshouse, 2003; Fleming *et al.*, 2005):

**Proximity** to the true Pareto-optimal front.

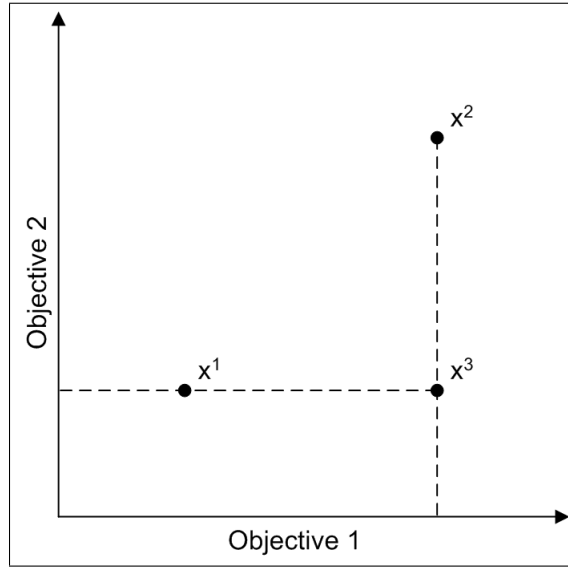


Figure 2.1: Illustration of Pareto-dominance assuming minimisation: solution  $x^1$  dominates  $x^2$ ,  $x^1$  dominates  $x^3$  and  $x^3$  dominates  $x^2$ .

**Diversity** between solutions on the Pareto-optimal front, *i.e.* sufficient distribution to reveal the underlying shape of the Pareto-optimal front and allow further search.

**Pertinency** of the Pareto-optimal solutions, *i.e.* they only lie in the decision-maker's (DM's) region of interest (ROI).

‘Classical’ search and optimisation methods have been used over at least the last four decades (Deb, 2001) to solve multi-objective optimisation problems. By ‘classical’, it is assumed that the method is deterministic and generates a single solution per iteration. This approach was dictated by the methods that were available at the time. A multi-objective problem was converted into a single objective problem using some sort of scalarising or aggregating function, based on the relative importance or preference of objectives. Examples include the weighted sum approach, as used in linear quadratic regulator design (Athans and Falb, 1966), and the goal attainment method (Gembicki, 1974). An extensive review of approaches is provided in Miettinen (1999). These methods require multiple runs of the optimiser to generate sufficient solutions to adequately represent the Pareto-optimal front. Furthermore, there is no guarantee with such approaches that multiple runs with various aggregations of objectives will produce the desired diversity in the Pareto-optimal front.

An alternative approach is the class of algorithms known as Evolutionary Algorithms (EAs). These originated in the 1970s (Rechenberg, 1973; Holland, 1975) and are based on the principles of biological evolution, *i.e.* ‘survival of the fittest’. They comprise a population of solutions, which is evolved or improved over a user-specified number of generations to produce the Pareto-optimal front in one run of the optimiser. Furthermore, the population approach can be used to emphasise all solutions equally and thus produce a diverse set of solutions. In addition, various preference articulation methods have been developed and applied to allow the DM to focus the population on the ROI. Multi-Objective Evolutionary Algorithms (MOEAs) are thus well suited to meet the aforementioned requirements of a MOO algorithm.

While there has been much research in the field of evolutionary multi-objective optimisation (Deb, 2001; Coello *et al.*, 2007), many of the applications have focussed on a relatively small number of objectives, *i.e.* often two or three. One obvious reason for this is the ease with which the resulting populations can be visualised using simple scatter plots to assist in selecting a preferred solution. However, real-world optimisation problems can comprise significantly more than two or three objectives and it was Farina and Amato (2002) who first introduced the term *many-objective optimisation* to describe such scenarios. (From hereon EMO will be used as the abbreviation for Evolutionary Many-objective Optimisation).

## 2.2 Issues in Many-Objective Optimisation

When trying to simultaneously optimise many objectives a number of difficulties arise:

- The ability of the optimiser to search towards the Pareto-optimal front can be compromised. This occurs when a large proportion of the population is non-dominated and there is insufficient selective pressure to progress the search (Khare *et al.*, 2003; Purshouse and Fleming, 2003b).
- The number of solutions, *i.e.* population size, required to approximate the Pareto-optimal front rises exponentially with the number of objectives, and thus can become prohibitive in terms of computational expense, as can be seen in Figure 2.2.
- Visualisation of solutions in many dimensions can be difficult. In addition, the

sparseness of such solutions in many dimensions due to the *curse of dimensionality* (Bellman, 1961) can make interpretation of the resulting hyper-surface a significant challenge.

Potential counter-measures are discussed in the following subsections. It should be noted that this review is not meant to be exhaustive. Rather, it summarises some recent and relevant methods and assesses their strengths and weaknesses. Other surveys of remedial approaches can be found in Ishibuchi *et al.* (2008); Zou *et al.* (2008).

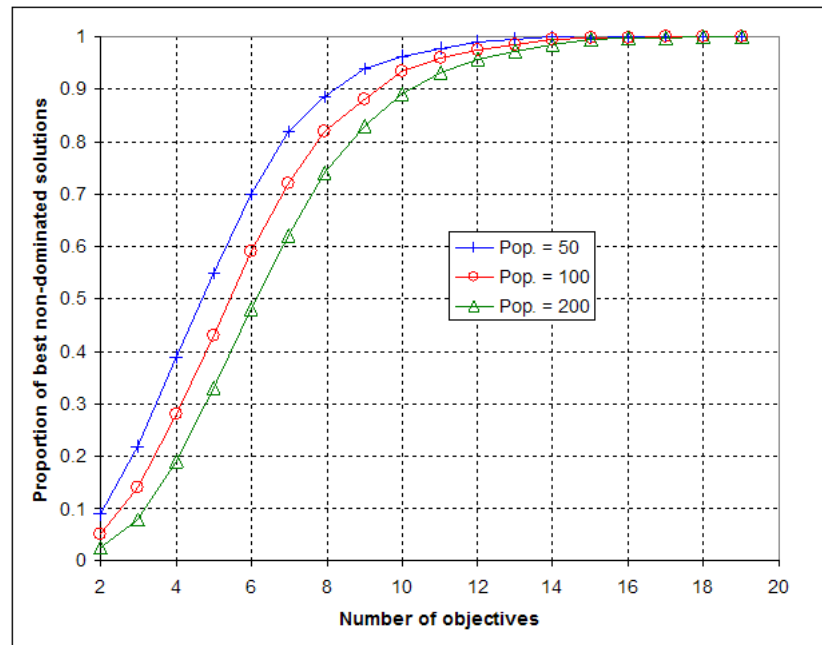


Figure 2.2: The proportion of non-dominated solutions versus the number of objectives for different population sizes, derived from data in Deb *et al.* (2003).

## 2.3 Search Efficiency

In this section an overview is provided of approaches to improve search efficiency in many-objective problems. These are categorised into: (i) methods for increasing selective pressure, (ii) alternative fitness functions, (iii) a convergence acceleration approach and finally (iv) parallel techniques.

In EMO problems, the lack of effective search, and hence convergence to the Pareto-optimal front, has been attributed to those algorithms based on Pareto-

dominance selection (Kukkonen and Lampinen, 2007), *e.g.* NSGAII (Deb, Pratap, Agarwal and Meyarivan, 2002) and SPEA2 (Zitzler *et al.*, 2001).

### 2.3.1 Methods for increasing selective pressure

#### Modification of Pareto-dominance.

The purpose of this approach is to reduce the number of non-dominated solutions in the population. In the approach of Sato *et al.* (2007), the dominance relation was changed to allow the user to contract or expand the dominance area. This was achieved with a user-specified scaling of the fitness value for each objective to weaken or strengthen selection. Another implementation was that of  $\alpha$ -dominance (Ikeda *et al.*, 2001) where a solution  $x$  dominates another  $y$  such that  $x$  can be slightly inferior (by an amount controlled by  $\alpha$ ) to  $y$  in one objective and superior to  $y$  in other objectives.

Both of approaches are examples of the application of weak dominance, which may increase the selective pressure and hence progress of the search towards the Pareto-optimal front. However, both researchers have observed that the diversity of the resulting population is reduced. In addition, both methods require the user to specify parameters to control the strictness of the dominance relation.

In Zou *et al.* (2004, 2008) *L-dominance* is defined. This is an extended form of Pareto-dominance taking account of the number and value of improved objectives. For most of the DTLZ problems (Deb, Thiele, Laumanns and Zitzler, 2002) tested with up to 9 objectives, it outperforms the state-of-the-art algorithms, IBEA (Zitzler and Künzli, 2004), MSOPS (Hughes, 2003), MSOPSII (Hughes, 2007) and NSGAII in terms of convergence to and diversity in, the Pareto-optimal front. However, it does rely on the additional task of normalising all objectives within the feasible set of solutions, *i.e.* the maximum and minimum of each objective must be found.

#### Assignment of different ranks.

One approach is that of Drechsler *et al.* (2001), which makes use of a relation *favour* to compare solutions. A solution  $x$  is *favoured* to another  $y$  if the number of objectives in which  $x$  is better than  $y$  is greater than the number of objectives for which  $y$  is better than  $x$ . A more sophisticated version of the relation *favour*, called  $\epsilon$ -Preferred, was developed by Sülflow *et al.* (2007) to incorporate a user-specified difference in objective



values,  $\epsilon$ , between compared solutions and so address a variability issue. This approach is not dissimilar to  $\alpha$ -dominance Ikeda *et al.* (2001).

Alternatively, Köppen and Yoshida (2007) consider different distance assignment functions to the existing crowding distance operator in NSGAII (Deb, Pratap, Agarwal and Meyarivan, 2002) to avoid the algorithm getting stuck in the first few generations. The main conclusion from this study is that the incorporation of substitute distance assignment functions in many-objective problems much improves the convergence to the Pareto-optimal front. However, this improvement is achieved at the expense of diversity in the population (since these functions are not explicitly designed to maintain diversity) and in some cases, higher computational expense.

Corne and Knowles (2007) compare different ranking methods and conclude that an average ranking method, such as Weighted Average Ranking (Bentley and Wakefield, 1997), outperformed other algorithms, except for problems comprising many objectives and where significant objective conflict exists. In addition, the performance of the methods tested is compared based on the *cover* metric (Zitzler *et al.*, 2003) only, which does not indicate convergence or diversity performance of the resulting population.

Ranking solutions and then aggregating them, described as *ranking-dominance*, is used to replace non-dominated sorting based on Pareto-dominance in Kukkonen and Lampinen (2007). Two aggregation functions are tested, the sum and the minimum of ranks, on the DTLZ problems on up to fifty objectives. For four of the six problems, ranking-dominance showed much improved convergence to the Pareto-optimal front. However, for the remaining two problems, ranking-dominance performed worse than Pareto-dominance. This was due to the fact that ranking-dominance generated solutions in which some objectives were deteriorated while others were improved.

Another observation was that in some cases, while convergence was much improved, diversity was significantly compromised, *e.g.* on a 10-objective DTLZ2 problem all two hundred individuals converged to just three unique solutions. To address this diversity issue, diversity maintenance based on a distance to the two nearest neighbours (Kukkonen and Deb, 2006) was gradually introduced across generations according to a power law. Results on the 10-objective DTLZ2 problem show that both good diversity and convergence have been simultaneously achieved with this approach. However, according to Kukkonen and Lampinen (2007), the author has limited this diversity

maintenance approach to one based on a power law, for one value of power and applied it to one test problem only. As a result, it is unclear how this method compares with alternative diversity maintenance techniques and how it performs on other test problems.

### 2.3.2 Different Fitness Functions.

As an alternative to Pareto-dominance, various fitness evaluation functions have been developed and applied. Indicator-based algorithms are one such approach, in which a single-objective indicator of a desired property of the population is optimised, *e.g.* the Indicator-Based Evolutionary Algorithm (IBEA) of Zitzler and Künzli (2004). A popular indicator is hypervolume, which measures the volume of objective space dominated by a population (Zitzler and Thiele, 1999). On selected many-objective DTLZ problems hypervolume used in an IBEA is shown to perform well (Wagner *et al.*, 2007).

However, it has the drawback of being computationally expensive and there have been various efforts to address this. Two examples are the iterative approach of Ishibuchi *et al.* (2007) and the fast hypervolume algorithm (HypE) of Bader and Zitzler (2008). The former only generates one solution per run and so multiple runs are required to generate a Pareto-optimal set of solutions. The latter proposes a fast search algorithm using Monte Carlo approximation of the hypervolume. When evaluated for a constant period of time, HypE outperforms popular MOEAs such as NSGAII in terms of hypervolume, albeit that each generation takes longer and hence it processes half the number of generations. The approximation of the Pareto-optimal front depends on the sampling of the Monte Carlo approach. More samples results in a more accurate approximation, but require a longer execution time.

Another fitness evaluation mechanism is that of scalarising or aggregation functions, which combine multiple objectives into a single objective, often with some form of weighting function. One such approach is Multiple Single Objective Pareto Sampling (MSOPS) due to Hughes (2003). It comprises multiple single objective searches run in parallel, each with a different aggregation of objectives using weight vectors. While this algorithm does not rely on Pareto-dominance to rank solutions and provide selective pressure, it does require specification of weight vectors *a priori*. This drawback is addressed by the MSOPS-II algorithm (Hughes, 2007), by incorporating automatic generation of weight vectors; although *a priori*-specified weight vectors gave superior

performance on some of the published test problems.

In Ishibuchi, Doi and Nojima (2006); Ishibuchi and Nojima (2007), a different scalarising function is used for each solution. Although this approach showed superior results to NSGAII (with Pareto-dominance) on the ‘knapsack’ problems tested, it also relies on the *a priori*-specification of weight vectors and probabilistic weighting between using the Pareto-dominance of NSGAII and a weighted sum formulation.

Two somewhat different alternative and relevant approaches are reviewed below. The former accelerates convergence to the Pareto-optimal front and could be used to address this issue in EMO. The latter, parallel MOEAs, is a whole field in itself and although related, has not been considered in any detail in the recent literature on improving EMO scalability.

### 2.3.3 The Convergence Acceleration Operator (CAO).

The approach taken by Adra *et al.* (2009) involves hybridising conventional MOEAs, NSGAII and SPEA2 with a convergence improvement process, where diversity maintenance is left to the existing mechanism in the host MOEA. At each generation the CAO comprises the following steps:

1. Using the existing population, generate a computationally cheap Radial Basis Function (RBF) network in objective space to predict the corresponding decision vectors. The parameter settings for the RBF and adaptive interpolation step  $h$  are generated from trial-and-error experiments. The parameter  $h$  is chosen to maximise local improvement in the Pareto-optimal front subject to remaining within the valid domain of the RBF. An adaptive process is proposed for setting  $h$  based on the proportion of solutions introduced by the CAO which propagate to the next generation.
2. A local improvement step via interpolation by a distance  $h$  of MOEA-generated solutions in objective space to improve the Pareto-optimal front.
3. A correction step, which firstly checks for any invalid decision vectors and changes them to valid nearest neighbours in decision space, and secondly, using the original objective functions, determines the *corrected* objective vectors from the predicted decision vectors in step 2. The use of this correction step was found to be critical

to the superior performance of the CAO-based NSGAII hybrid compared to the standard NSGAII. Together with the fact that a new RBF is created at every generation, this correction step goes some way towards addressing difficulties with mapping from objective to decision space.

4. Finally, the new objective vectors from step 3 then compete with those in the MOEA archive to generate a new, fittest population.

Improved convergence compared to NSGAII and SPEA2 was observed on ZDT (Zitzler *et al.*, 2000), DTLZ and real-world problems. Although the inclusion of the CAO into MOEAs increases the computational expense per generation, fewer generations are needed to achieve similar performance. The CAO-based NSGAII hybrid generated superior results to NSGAII for a similar computational effort.

#### 2.3.4 Parallel MOEAs.

Historically, parallel or distributed computing has been an important initiative in solving time-consuming real-world optimisation problems. As previously mentioned, the proportion of non-dominated solutions in the Pareto-optimal front becomes large as the number of objectives is increased, as shown in Figure 2.2, and the selection pressure correspondingly reduces. In addition, in order to generate a diverse Pareto-optimal front a large population is required, which can be computationally expensive with serial Multi-Objective Evolutionary Algorithms (MOEAs), but may be much less time-consuming with parallel MOEAs (pMOEAs).

##### pMOEA paradigms.

Implementations of pMOEAs can be classified into three different approaches as listed below with a much fuller description and many applications in Coello *et al.* (2007):

- **Master-slave model.** In this paradigm objective functions are evaluated on several slave processors, while, at each generation, the master processor carries out genetic operations and other functions such as: collecting and distributing sub-populations, determining the non-dominated front and archiving. Computational acceleration is only realised if the objective functions are expensive to evaluate, in which case computation time is much greater than communication time between the master and the slaves.

One recent example of this approach is that of Shenfield *et al.* (2007), where expensive objective functions were farmed out to slave processors connected in a computational grid. Remote interaction with the master processor allowed the optimisation to be steered towards a ROI. For the research undertaken in this Ph.D. programme, response surface models, with execution times in milliseconds, represent the objective functions. Consequently, the master-slave model is not considered applicable.

- **Island model.** This is analogous to a chain of islands, each with its own population, where limited migration occurs between islands. The pMOEA population is divided into a number of independent sub-populations or *demes*, each of which runs on a separate processor with an independent MOEA. Periodically, a small number of best non-dominated individuals are migrated to neighbouring islands (processors).

A suitable migration policy involves defining the number of migrants, the migration frequency, and the migrant selection and re-insertion method. This paradigm is well suited to distributed computing using a cluster of processors, as exists for this research, because there is limited communication between the host pMOEA and the island or worker MOEAs.

- **Diffusion model.** This approach is similar to the master-slave paradigm, except that each processor operates on only one or a few individuals in a small local neighbourhood. The aim is that good solutions emerge from each neighbourhood and slowly *diffuse* throughout the population. This requires a massively parallel computer, which was not readily available for this research.

In conclusion, due to the computing resources available, *i.e.* a compute cluster, and the execution speed of the objective functions used, *i.e.* very fast, the only suitable approach that could be applied to this research is an island-based pMOEA.

#### pMOEA implementation aspects.

Given this assumption, there are various implementation aspects to be considered:

- **Niching.** The concept of allocating individual sub-populations to concentrate on optimising a particular portion of the Pareto-optimal front is appealing as this

may be more efficient in generating a diverse population. There have been several implementations of this idea including:

- At regular intervals the Divided Range Multi-Objective Genetic Algorithm (DRMOGA) (Hiroyasu *et al.*, 2000) gathers sub-populations, sorts according to a temporarily specified objective (this changes for each sort), and then redistributes onto the island processors. The drawbacks of this process are that the choice of objective on which to sort is arbitrary, the sorting is temporary and the communication overhead is significant (Branke *et al.*, 2004).
- In Deb *et al.* (2003) the Guided Domination approach (Branke *et al.*, 2001) is used to guide the optimisation to different parts of the Pareto-optimal front. Although this was shown to be effective on the problems tested, the published approach is only applicable to convex Pareto-optimal fronts and furthermore, *a priori* knowledge of the Pareto-optimal front is required to specify the search directions.
- In Branke *et al.* (2004) the search space is divided into a number of regions each assigned to a different processor. This sub-division is achieved by applying a cone constraint whose vertex is at the nadir point (worst point in all objectives) resulting in an equal partition per processor. Unless the Pareto-optimal front is known *a priori* this approach requires the nadir point to be determined. While the efficacy of the method was demonstrated on 2-objective test problems, when tested on a 3-objective problem there was a lack of diversity in the resulting population as evidenced by a concentration of solutions at the borders between regions.
- Another ‘divide and conquer’ approach is that of Streichert *et al.* (2005) where *k*-Means clustering is applied to sub-divide the population among *np* processors. At some migration rate, the sub-populations are collected, clustered and re-distributed. This algorithm was tested on a number of problems, albeit all 2-objective, and showed that the standard island approach with migration (at a rate of 5 individuals every 2 generations) gave equivalent performance. As with DRMOGA, there will also be a communication expense to this partitioning process, which may be significant at such

a migration frequency.

- **Migration and Replacement.** Many migration and replacement schemes exist and these are categorised and described in detail in Coello *et al.* (2007). Two migration aspects of interest are the number of immigrants and the processor destination for immigrants. If too many individuals are migrated there is a risk that identical sub-populations will be generated. Conversely, if too few individuals are migrated then the search of objective space may not be effective. With regard to the destination processor, if no migration takes place then ‘gene mixing’ cannot take place and there is the risk that the local MOEA may converge prematurely. Conversely, a one-to-all processor broadcast of immigrants could cause an excessive communication overhead.

Multiple replacement schemes also exist. The simplest approach and that with the lowest selective pressure is the *random* method where individuals are randomly replaced with immigrants. Conversely, *elitist 100% ranking* provides the greatest selective pressure. Here immigrants are added to the current population, the resulting population ranked into non-dominated fronts and then individuals are deleted from the worst fronts. The drawback with *elitist 100% ranking* is the computational cost of ranking the population into fronts.

- **Archiving.** Maintaining an external database of best individuals, which is periodically combined with the local (island) population has the potential to improve pMOEA search efficiency. A suitable balance should be found between regular archive update, the number of archives, the method by which they are maintained and the communication overhead. One straightforward implementation is to allow local archiving and at the end of the pMOEA search, all the local archives are combined to determine the final Pareto-optimal front.

## 2.4 Visualisation

Much research on evolutionary multi-objective optimisation has concentrated on two-objective problems Deb (2001); Coello *et al.* (2007). Using a Cartesian scatter plot, it is straightforward to visualise the optimisation results and select a preferred non-dominated solution. However, when there are more than two objectives, as is often the

case in engineering problems, the task of presenting sufficient, but not excessive, visual information to the DM (Decision Maker) for decision-making purposes to be made, is more challenging. Two commonly-used approaches are reviewed below with more extensive surveys in Miettinen (1999); Deb (2001). This is followed by a brief review of some recent methods.

### 2.4.1 The Scatter-Plot

The scatter-plot as described in Cleveland (1993) is in fact a matrix of pairwise objective plots. A real-world (automotive) example of a Pareto-optimal front is shown in Figure 2.3. Despite widespread usage of such plots, particularly for the results of two-objective problems, they can become increasingly difficult to interpret for higher numbers of objectives. In other words, visually representing three or more dimensions with a two dimensional diagram may not be very informative to the DM. While projecting onto lower dimensional spaces may seem appealing, it carries with it the risks of losing information and in particular, hidden extrapolation, with regard to valid model domains. Nevertheless, a scatter plot matrix combined with *brushing* (highlighting one or more objective vectors) (Wegman and Luo, 1997), as implemented in Matlab<sup>®</sup> v7.7 (R2008b), can be useful for simultaneous highlighting of solutions of interest in multiple

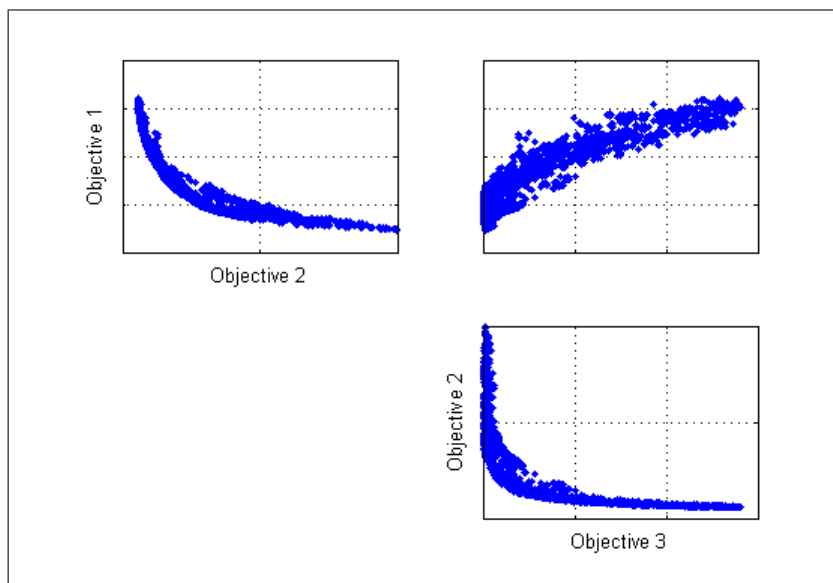


Figure 2.3: Scatter plot matrix of a Pareto-optimal front from a real-world (automotive) problem



dimensions.

### 2.4.2 The Parallel Coordinates Plot

The parallel coordinate representation as originated by Inselberg (1985), and in the case of a multi-objective evolutionary optimisation, involves plotting the normalised objective components of the resulting individuals onto parallel axes, one per normalised objective. The objective components for each individual are joined by a line. The degree to which the lines cross indicates how strong the trade-off is between adjacent objectives, *i.e.* the more/less the lines cross, the more/less the adjacent objectives conflict. An example plot is shown in Figure 2.4 using the same data as in Figure 2.3. However, care should be taken in interpreting such plots, as changing the axis scaling can dramatically affect the appearance of any pairwise objective correlation. Clearly, the ordering of the objective axes is important to reveal the presence and degree of any conflict. The ability to easily re-order or re-scale axes can substantially change the view of the resulting objective space and offer new insights into the data structure (Müller *et al.*, 2008).

Although this method has not enjoyed the same substantial experience and intuition that has been gained with Cartesian plots (Wegman, 1990), parallel coordi-

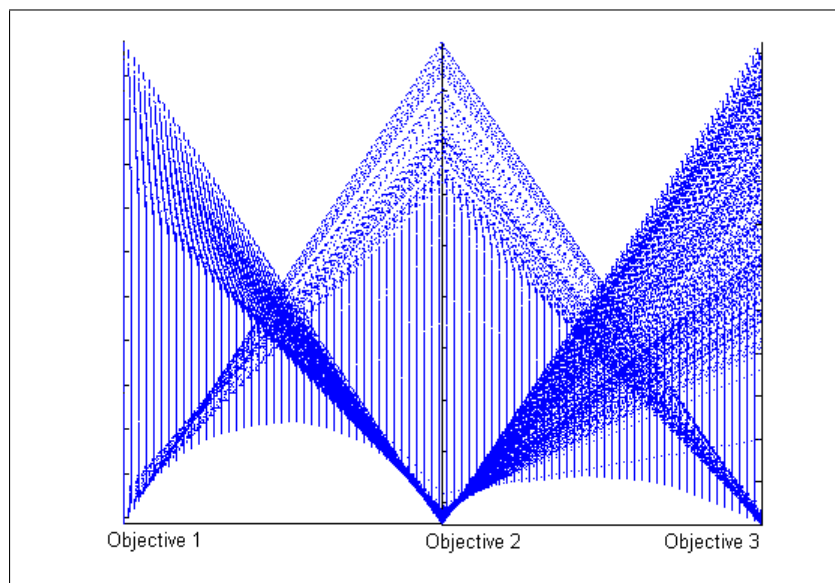


Figure 2.4: Parallel coordinates plot of a Pareto-optimal front from a real-world (automotive) problem (same data as in Figure 2.3)

nates plots do allow a multi-dimensional objective space to be represented in a two-dimensional diagram. Wegman and Luo (1997) describe further development of the use of parallel coordinates with the introduction of the Grand Tour (Asimov, 1985) methodology, which animates two-dimensional projections of high-dimensional data to reveal the underlying data structure.

Example applications of the use of parallel coordinates are the interactive multi-objective optimisations shown in Fonseca and Fleming (1998b) and Fleming *et al.* (2005). Here goals are overlaid on the parallel coordinates plots to assist the DM to progressively articulate preferences, so as to reduce the region of interest and ultimately identify an acceptable solution.

Schroder (1998) developed a method of quantifying the extent of trade-off or conflict between objectives, which originated with parallel coordinates plots. Each objective was partitioned into regions, in each of which a measure of conflict was calculated based on the weighted sum of the separation of pairs of objectives, where a larger separation is indicative of greater conflict. The measure of conflict is normalised by population density. One shortcoming of this approach and one that in fact applies to all pairwise comparison approaches, is that there may be conflict between three or more objectives for a particular solution.

### 2.4.3 Recent Visualisation Approaches

In Blasco *et al.* (2008), Pareto-optimal solutions are classified according to their normalised proximity to an ideal (or utopian) point. A weakness of this approach is that the minimum and maximum of each objective function needs to be determined, which may not be known *a priori* and knowledge of which may change in EMO problems as the search progresses. Nevertheless, a useful feature presented is that of colouring the Pareto-optimal solutions according to their objective preference.

Yoshikawa *et al.* (2007) makes use of Fuzzy C-Means clustering to partition the Pareto-optimal front. Subsequently, for each cluster, Fuzzy Multiple Discriminant Analysis is used to identify projection axes (or eigenvectors), in a similar way to principal component analysis. The sign and magnitude of the resulting eigenvector coefficients is used to reduce objectives, although the process for doing this is unclear and does not seem to be part of a progressive objective reduction process. A weakness of the clustering method used is that the initial number of clusters must be specified.

A Hyper-Space Diagonal Counting method is applied in Agrawal *et al.* (2004). This approach relies on dividing each objective range into bins and counting the solutions falling into each bin. For more than two objective problems, the objectives are grouped so that the results can easily visualised in two or three dimension scatter plots. Both the optimal division of each objective into bins and the most appropriate grouping of objectives are unresolved issues with this method as pointed out by the authors. Furthermore, when the objectives are grouped, it is not clear how to separate solutions which fall into the same bin, but have significantly different objective values.

## 2.5 Pertinency

Preferences can be used to confine the optimiser to certain regions of interest of the trade-off surface. Various approaches used in Multi-Criteria Decision Making (MCDM) include the specification of constraints on decision variables and/or criteria (objectives), relative importance of criteria and comparison of solutions, as well as aspirations or goals for the criteria. Four categories of decision making processes appear in the literature (Hwang and Masud, 1979) (see surveys in Coello (2000); Andersson (2000); Miettinen (2001, 2002); Fleming *et al.* (2005); Rachmawati and Srinivasan (2006); Adra *et al.* (2007)). One category, that of *no preference articulation*, comprise methods which do not make use of any preference information, but merely generate optimal solutions (Yu, 1973; Zeleny, 1973; Miettinen and Mäkelä, 1995). Since a decision making stage is considered essential in real-world EMO problems, such methods will not be explored further. The remaining three categories are described below:

### 2.5.1 *A priori* Preference Articulation

This is when the DM specifies preferences prior to the optimisation. A common approach is to specify an aggregating function, which converts individual objectives into a single utility function resulting in a single objective optimisation problem. An example of this approach is the weighted-sum method (Hwang and Masud, 1979). Here, each of the objectives is multiplied by a weight and then all the weighted objectives are aggregated to produce a single objective function. Other *a priori* approaches include the Goal Programming method of Ignizio (1976), the utility function approach employed by Greenwood *et al.* (1997) and the weighting function approaches used by Bentley

and Wakefield (1997). Although these approaches possess the virtue of simplicity, they have some well known difficulties with (Deb, 2001; Purshouse, 2003):

- Identifying non-convex Pareto fronts.
- Generating a well-distributed set of Pareto-optimal solutions.
- Precise capture of the decision maker's preferences in a single utility function.
- Aggregating non-commensurable objectives.

Furthermore, such approaches are capable only of finding a single optimised solution in a single simulation run. Due to these difficulties and since *a priori* methods are essentially single objective approaches, they will not be considered further as a preference articulation method.

### 2.5.2 *A posteriori* Preference Articulation

Here the DM specifies preferences after the optimisation to identify a preferred solution. Examples include Pareto optimisation approaches NSGA (Srinivas and Deb, 1994), MOGA (Fonseca and Fleming, 1998a), and SPEA (Zitzler and Thiele, 1999), non-Pareto approaches such as VEGA (Schaffer, 1985) and VOES (Kursawe, 1991), as well as other EAs such as MOSA (Suppaitnarm *et al.*, 1999), and gradient-based methods such as NBI (Das and Dennis, 1998). Historically, much EMO research has concentrated on *a posteriori* approaches and neglected to consider the subsequent decision-making phase, which is often of great importance in real-world applications. When applied to EMO problems, common issues with *a posteriori* methods are:

- The algorithm may have some difficulty in generating an adequate Pareto front in terms of its diversity of solutions and proximity to the true Pareto front (Purshouse, 2003).
- It may be computationally infeasible for the algorithm to generate Pareto optimisations for many objectives, particularly if in the case of evolutionary algorithms, large populations are used. As the DM is usually only interested in a subset or region of interest (ROI) of the Pareto front, then many of the Pareto-optimal solutions may be redundant.

### 2.5.3 Progressive Preference Articulation

This category is progressive or interactive articulation of preferences. In this group, the DM iteratively pauses the optimisation to provide revised preferences as more information progressively becomes available and then resumes the optimisation to search for better solutions (Cvetković and Coello, 2004; Drechsler *et al.*, 2001; Branke *et al.*, 2001; Tan *et al.*, 2003; Branke and Deb, 2004; Fonseca and Fleming, 1998a). Thus, the DM acquires knowledge of the problem as the optimisation progresses and very little pre-requisite information such as, goal values, objective ranges and individual objective optima is required. Monarchi *et al.* (1973) gave several reasons for favouring progressive preference articulation, which include:

- Preferences cannot be quantified analytically, but the DM does subscribe to a certain view.
- Solutions and preferences may change over time.
- Goals and preferences change as solutions emerge and experience of the problem is gained.

Miettinen (2001, 2002) provides an extensive survey of interactive multi-objective optimisation methods, but all those described scalarise the objectives into a single objective providing a single solution. Since knowledge of the Pareto-optimal front – as defined by a family of solutions – is necessary information for the DM to specify preferences and guide the optimisation, scalarising or aggregating or single objective approaches will not be considered further. Where multiple Pareto-optimal solutions are generated a progressive process can be used to ‘zoom in’ on the ROI on the Pareto front. Compared to *a posteriori* methods, by virtue of searching in a smaller objective space, this approach should provide both:

- More and better distributed Pareto-optimal solutions in the ROI for the same computational expense.
- Better proximity to the true Pareto front.

Hence, progressive preference articulation (PPA) represents a promising technique for finding Pareto-optimal solutions in EMO problems. Example approaches include

Cvetković and Coello (2004); Drechsler *et al.* (2001); Branke *et al.* (2001); Tan *et al.* (2003); Branke and Deb (2004); Fonseca and Fleming (1998a).

## 2.6 Harmony, Conflict and Independence

### 2.6.1 Definitions

As previously described in Purshouse (2003) and Deb and Saxena (2006), there exists an opportunity to reduce the dimensionality in many-objective problems if, for the Pareto-optimal solutions in the DM's region of interest, objectives are sufficiently positively correlated, *i.e.* in harmony. In this case, improvement in one objective would automatically improve another positively correlated objective. As a means of explaining this refer to Figure 2.5, which is a graph of several Pareto-optimal solutions,  $x_1, \dots, x_5$  plotted against two objectives,  $a$  and  $b$ , both of which are minimised. Solutions are compared pairwise and relative to  $x_1$  to define the nature of the dependency.

- **Harmony.** In these regions solutions  $x_2$  and  $x_4$  are compared to  $x_1$ . From the definition of harmony given in Purshouse and Fleming (2003a),  $(x_1^a < x_2^a) \wedge (x_1^b < x_2^b)$ , likewise with  $x_4$  compared to  $x_1$ . For  $x_1$  and  $x_2$  (and for  $x_4$  and  $x_1$ ) complete

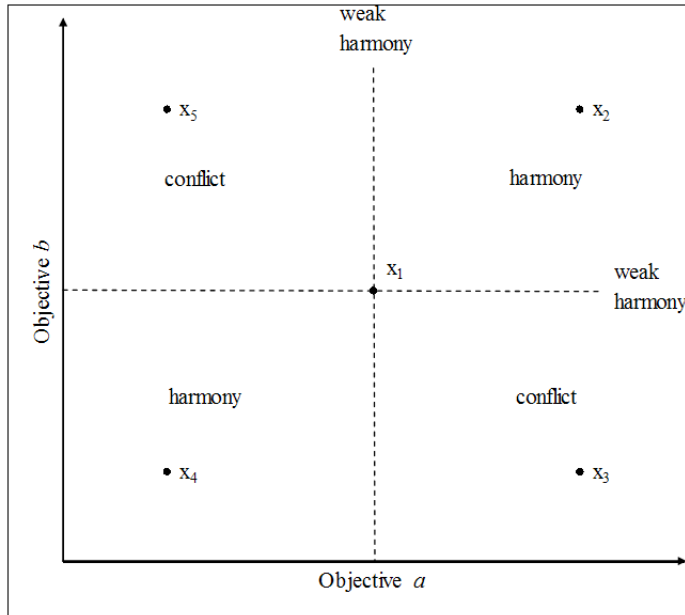


Figure 2.5: Conflict and harmony between two solutions with respect to two objectives  $a$  and  $b$ , both of which are minimised.

harmony (or complete correlation) would exist if the gradient of the line joining both solutions is 1; in other words, the correlation between the two objectives (for these two solutions) is 1. Weak harmony would exist if  $(x_1^a < x_2^a) \wedge (x_1^b = x_2^b)$  or  $(x_1^a = x_2^a) \wedge (x_1^b < x_2^b)$ . By extension, if  $x_4$ ,  $x_1$  and  $x_2$  were the Pareto-optimal set, then objectives  $a$  and  $b$  would be in harmony.

- **Conflict.** In these regions, solutions  $x_3$  and  $x_5$  are compared to  $x_1$ . Again from Purshouse and Fleming (2003a), conflict is defined as  $(x_1^a < x_3^a) \wedge (x_1^b > x_3^b)$  and likewise with  $x_5$  compared to  $x_1$ . For  $x_3$  and  $x_1$  (and for  $x_5$  and  $x_1$ ) complete conflict would exist if the gradient of the line joining both solutions is -1; in other words, the correlation between the two objectives (for these two solutions) is -1. Again, this definition can be extended so that if the Pareto-optimal set comprised solutions,  $x_5$ ,  $x_1$  and  $x_3$ , objective  $a$  would be in conflict with objective  $b$ .
- **Independence.** In the case where the solutions compared do not form part of the same Pareto-optimal front, *e.g.* in different sections of a disconnected Pareto-optimal front, then the objectives are perfectly independent for such solutions. In this case, the correlation is 0 and the objectives can be minimised independent of each other.

### 2.6.2 Global and Local Harmony and Conflict

Both Purshouse and Fleming (2003a) and Deb and Saxena (2005) state that harmony and conflict may vary across the Pareto-optimal front. This is depicted in Figure 2.6 and comprises a plot of Pareto-optimal solutions (hollow circles) overlaid onto the Pareto-optimal front (mesh) for the three-objective DTLZ2 problem (Deb, Thiele, Laumanns and Zitzler, 2002) where all objectives were minimised. It can be seen from the bottom right that  $z_1$  and  $z_3$  are in harmony and  $z_2$  is in conflict, whereas at the top  $z_1$  and  $z_2$  are in harmony, but  $z_3$  is in conflict. One potential issue, as pointed out by Purshouse (2003), is that if a redundant (*i.e.* positively correlated) objective is left in the problem, then the distribution of solutions on the Pareto-optimal front may be biased towards such objectives. This effect is a function of how effective the diversity preservation mechanism is within the genetic algorithm.

In summary, it has been recognised that harmony and conflict vary locally within the Pareto-optimal front, which in turn presents an opportunity for local objec-

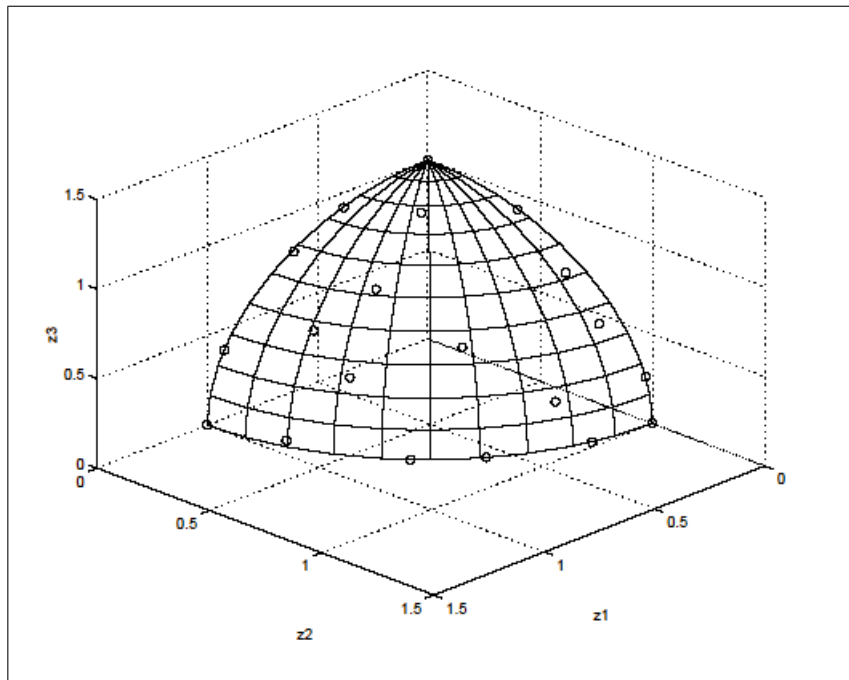


Figure 2.6: DTLZ2 scatter plot showing how conflict varies across the Pareto-optimal front from Purshouse and Fleming (2003a).

tive reduction. The author is not aware of any research which exploits this opportunity for dimension reduction in EMO problems.

## 2.7 Dimension Reduction

Clearly, if it is possible to reduce the dimensionality of a many-objective problem then this is desirable to improve the effectiveness of the optimisation algorithm in locating the true Pareto-optimal front. In identifying dimension reduction as a remedial measure, Purshouse and Fleming (2003a) stated that such approaches can be used interactively or *a priori* and briefly described two examples namely, Curvilinear Component Analysis (Demartines and Hérault, 1997) and Principal Component Analysis (Kendall, 1975; Jolliffe, 2002). These techniques are amongst several that have received widespread development and application in fields such as multi-variate statistics, image processing, engineering, astronomy and biology.

In such applications it has been found that some of the variables are correlated with each other and thus the dimensionality can be reduced by finding a subset of uncorrelated variables to generate a more compact representation of the data. The



need for analysing very large multi-variate data sets, which would otherwise prove intractable, is in no small part due to the extensive computing and storage capabilities now available to many institutions. Some dimension reduction approaches are reviewed below with more complete surveys described in Carreira-Perpinan (1997) and Fodor (2002). It is convenient to categorise the various methods into linear and non-linear methods.

### 2.7.1 Linear Methods

The two most widely used methods are principal component analysis (PCA) (Kendall, 1975; Jolliffe, 2002) and Factor Analysis (Harman, 1967; Kendall, 1975). These techniques fall into the category of second-order methods, *i.e.* rely only on information contained in the covariance matrix. As such, these methods often only require classical matrix manipulations and so are computationally straightforward.

- **Principal Component Analysis.** This is also known in some fields as the singular value decomposition (SVD) and the Hotelling transform (Hotelling, 1936). The basic principle behind PCA is to reduce the dimensionality of the problem by finding the orthogonal linear combinations (or Principal Components) of the original variables that account for most of the variance in the data. Furthermore, it can be used to identify any near-linear dependencies and thus immediate objective reduction. In other words, PCA exploits the existence of linear associations or dependencies between variables (for the purposes of this research, the term *objectives* replaces the general term, *variables*).

Deb and Saxena (2005) applied PCA to reduce the dimensionality of EMO problems with up to 30 objectives. Procedures were proposed to identify from the whole population the significant principal components and then to reduce the number of objectives. Standard test problems were used with known solutions. The authors demonstrated that this methodology has some vulnerability in finding the Pareto-optimal front in a 10-objective problem.

Brockhoff and Zitzler (2006) questioned whether the PCA approach of Deb and Saxena (2006) reduced the number of objectives without consideration of the effect on the dominance structure. This concern can be addressed by considering a straightforward example. If objective 1 and objective 2 are strongly positively

correlated (or harmonious), then objective 2 can be eliminated because it is redundant. It is still true that if the remaining objective 1 is subsequently improved then this would have simultaneously improved objective 2 if it had been left in the problem (this, of course, assumes that the search is constrained to remain in the same region of the Pareto-optimal front, otherwise such correlations may change). So, in the case of strongly positively correlated objectives, the relevance of maintaining the dominance structure is not clear. What is important is retaining any conflicting objectives in the DM's ROI. Nevertheless, the approach of Brockhoff and Zitzler (2006, 2009) is of interest in that exact and heuristic algorithms are developed and demonstrated for determining the *minimum objective subset* and the *minimum objective subset of given size with minimum error* (in the dominance structure). Another objective reduction method is that of Jaimes *et al.* (2008), in which conflict, based on distance between solutions in objective space, and an objective correlation matrix are used to discard the least conflicting objectives. Deb and Saxena (2005); Brockhoff and Zitzler (2006) and Jaimes *et al.* (2008) approach objective reduction by identifying global conflict, *i.e.* across the whole Pareto-optimal front, rather than locally in regions of it.

- **Factor Analysis.** This assumes that the measured variables depend on some unknown, fewer common (or latent) factors. This dependency is defined by a linear model, which includes an error or noise term. This differs from PCA where no explicit model is defined. In addition, it is mandatory in Factor Analysis to search for a rotation of the factors to give a simpler interpretation of the relationship between the variables and factors. This rotation takes the form of an orthogonal or oblique transformation to estimate the model parameters or factor loadings.
- **Higher Order Methods.** These apply to non-Gaussian data sets. One such technique is **Projection Pursuit** (Friedman and Tukey, 1974), which involves searching for *interesting* projections that optimise a projection index, where the Gaussian distribution is considered the least interesting. However, Projection Pursuit is not well suited to highly non-linear data and is computationally intensive (Carreira-Perpinan, 1997). **Independent Component Analysis** (ICA) (Hyvärinen, 1999) is another higher order method, which identifies linear trans-

formations which are as independent as possible. However, it relies on some other method, *e.g.* PCA, to reduce the dimensionality beforehand.

### 2.7.2 Non-Linear Methods

- **Non-Linear PCA.** One approach extends the vector of variables,  $\mathbf{x}$ , to include linear functions of variables that have maximum variance (Gnanadesikan, 1977). For example,  $\mathbf{x} = (x_1, x_2)$  could be extended to  $\mathbf{x}' = (x_1, x_2, x_1^2, x_2^2, x_1x_2)$ . In addition, Jolliffe (2002) describes other non-linear transformations such as products of powers of the variables. Obviously, it is important that whatever transformation is applied, it holds for all the data under consideration. One recent application of non-linear PCA, in particular Kernel PCA, is described in Saxena and Deb (2007), but the success of this approach relies on the choice of kernel.
- **Non-linear Principal Curves.** These are smooth one-dimensional non-linear curves which pass through the *middle* of multidimensional data such that the variation orthogonal to the curve is minimised. Principal Surfaces are principal curves extended to multiple dimensions (Hastie and Stuetzle, 1989). However, the required smoothing is difficult if data is sparse in multiple dimensions. Furthermore, it is not known for what kinds of distribution principal curves exist and if the construction algorithms converge (Carreira-Perpinan, 1997).
- **Multi-Dimensional Scaling (MDS).** This method maps high dimensional data onto low dimensional space based on a measure called *proximity*, which indicates the similarity (or correlation) between pairs of objects. It is commonly used prior to cluster analysis. Shortcomings of MDS include determining the dimension of the lower dimensional map - this is a process of trial-and-error and unlike PCA, it is not possible to generate a lower dimensional MDS map from a higher dimensional map (Carreira-Perpinan, 1997; Morrison *et al.*, 2003). One form of MDS is that of Maximum Variance Unfolding as described in Saxena and Deb (2007), where Semi-Definite Programming is used to maximise the proximity matrix with a constraint on the number  $nn$  of nearest neighbours, with whom distances and angles have to be maintained in the process of unfolding. Saxena and Deb (2007) state that the choice of  $nn$  is crucial to avoid over-constrained or erroneous unfolding, but do suggest a guideline.

- **Topologically Continuous Maps.** These are of fixed dimension and are generated from high dimensional data using unsupervised learning, the most well known of which are the Self-Organising Maps (SOMs) of Kohonen (1995). SOMs have been used in conjunction with MOEAs (Obayashi and Sasaki, 2003; Obayashi *et al.*, 2005) to give an initial clustering of design objective functions prior to hierarchical clustering of ‘adjacent’ clusters. However, due to the heuristic nature of SOMs they have a number of disadvantages including the lack of a cost function to optimise and no guarantee of convergence (Carreira-Perpinan, 1997). Furthermore, in order to make such representations visually intuitive, it is necessary to consider visual enhancements such as shading, colour, the annotation of design solutions and legends. Also, the training of the SOM needs to be computationally fast so that the decision-making process is efficient.
- **Vector Quantisation.** Kambhatla and Leen (1997) used a hybrid approach of vector quantisation to cluster the data into disjoint regions, within which a local PCA is conducted to achieve a locally linear dimension reduction by retaining the *rpc* Principal Components where  $rpc < n$  where  $n$  is the dimension of the data. The vector quantisation algorithm used minimises the reconstruction error to achieve an optimal dimension reduction. However, no guidelines are provided for selecting the target dimension *rpc*.

## 2.8 The Research Gap

Based on the review of the relevant literature in the previous sections, a proposed *practical many-objective optimisation decision-making process* (or practical MOODM process) would have the following features:

- Be able to efficiently generate trade-off solutions useful to the DM in his/her region of interest. In the field of automotive powertrain calibration, more stringent legislation, more demanding customers and intense competition has lead to increasingly complex powertrain technology. Such considerations often lead to more competing objectives and constraints, which need to be efficiently evaluated to define optimal system capability and generate optimal calibrations.
- In the DM’s region(s) of interest, identify and exploit local objective harmony for

potential objective reduction. That is, be able to partition the Pareto-optimal front into domains of like solutions and then within each domain, determine objective dependencies and discard harmonious objectives (from subsequent optimisations, albeit that the DM should be able to check that any objective harmony is maintained). This objective reduction process should be straightforward and efficient to perform for the DM, who is likely to be an engineer unfamiliar with the details of the process.

- Utilise progressive preference articulation (PPA) to allow the DM to iteratively specify objective goals and priorities at each stage of the optimisation, as more information about the problem becomes known. Preference articulation allows the DM to specify a region of interest and hence shrink the search space and also to discriminate between harmonious objectives, which may be discarded. The approach employed should be able to work with the current generation of MOEAs, *e.g.* NSGAII (Deb, Pratap, Agarwal and Meyarivan, 2002), allow reformulation of the optimisation problem to cater for discarded objectives and again, be intuitive and efficient to use for the DM. Although there is evidence that NSGAII's search ability deteriorates in many objective problems (Hughes, 2005; Ishibuchi, Nojima and Doi, 2006), it is hoped that the addition of PPA to shrink the search space will mitigate this concern (Bentley and Wakefield, 1997). This remains an opportunity for future research.
- Be a systematic and as far as possible, objective (rather than subjective) process that progressively reduces objective dimension and arrives at a preferred Pareto-optimal solution per local region of the Pareto-optimal front.
- This process should be modular, so that each component process can be improved as better techniques emerge so as to leave the overall process intact.
- Be as computationally efficient as possible given that many-objective problems by definition have a larger dimension in objective space. With MOEAs this may necessitate large populations, many generations and with progressive approaches, multiple optimisation stages. Hence, the partitioning of the Pareto-optimal front and objective reduction process should also not impose a significant time burden on the DM.

In view of the above requirements of the proposed process, it is useful to summarise and highlight weaknesses of existing approaches and hence research opportunities as reviewed previously in this chapter.

- To determine calibration trade-offs amongst competing engine responses parametric studies have been utilised (Deb, 1995; Montgomery and Reitz, 1996), which are time-consuming and have no guarantee of locating Pareto-optimal solutions. Multi-objective optimisation algorithms coupled with mathematical (empirical and/or physical), fast engine simulations provide a means to systematically search for these Pareto-optimal solutions. Multi-objective genetic algorithms have proved to be effective search methods for problems with small numbers of objectives, but for more than three objectives such algorithms can suffer from insufficient selective pressure to enable progress to the true Pareto front (Ishibuchi *et al.*, 2008).
- Although exploitation of objective harmony for dimension reduction has been demonstrated (Deb and Saxena, 2005), this has been applied on the whole population. In other words, this approach has identified evidence of global harmony and conflict. However, it has been established (Purshouse, 2003; Deb and Saxena, 2006) that objective harmony and conflict vary across the Pareto front, *i.e.* that *local* harmony and conflict exist. To the best of the author's knowledge the only known local objective reduction approach is that of Yoshikawa *et al.* (2007). However, as detailed in Section 2.4.3, it has a number of weaknesses including an unclear objective reduction process, which does not appear to be progressive, and in addition, relies on the initial number of clusters being specified.
- With regard to linear dimension reduction approaches, Factor Analysis assumes that there is some random error in the data being analysed when there is no such component in the mathematical models, which are evaluated to generate solutions. By contrast, PCA does not make such an assumption, but is cited as not being suitable for non-linear data such as that typical of Pareto-optimal fronts. However, if the Pareto-optimal front is partitioned into groups of like solutions, then PCA may be useful in identifying local harmony for objective reduction. Higher order methods which allow for non-Gaussian data, can be

computationally expensive or may rely on other methods, *e.g.* PCA (Carreira-Perpinan, 1997; Hyvärinen, 1999).

With non-linear methods, the DM may need to specify additional information such as the non-linear transform required or distributional assumption. In addition, there are known problems with MDS not being able to project onto lower dimensions; with SOMs, which have issues with subjectivity involved in hierarchical clustering, convergence and interpretation; and with Vector Quantisation, where the DM must specify target dimension *a priori* and no consideration is given to objective harmony and conflict (Kambhatla and Leen, 1997).

- Of the two common visualisation approaches, scatter plots are difficult to interpret for more than three objectives and parallel coordinates plots, while useful as a one-dimensional representation regardless of the data dimension, can be difficult to interpret for large data sets and may require all objective orders to be plotted to reveal any pairwise objective dependencies. Nevertheless, parallel coordinates plots have been useful in a PPA context, where objective goals can be overlaid to aid decision making (Fonseca and Fleming, 1998b; Fleming *et al.*, 2005).
- Preference handling approaches also have limitations. *A priori* preferences are not usually known before the optimisation starts, while *a posteriori* approaches run into difficulties with many objectives in terms of effective search (Fleming *et al.*, 2005). There are various PPA approaches, with that of Fonseca and Fleming (1998a) being one of the more intuitive to use as objective priorities and goals can be specified fairly simply and directly. In addition, it has been shown to perform well compared to other recent PPA methods (Adra *et al.*, 2007). However, to date this approach has not received widespread application.

## 2.9 Review of Relevant Automotive Powertrain Technologies

Case studies are used in the body of the text to demonstrate the utility of the proposed methodology developed in this thesis. Both studies are automotive applications. For convenience, background information pertinent to each study is presented here, so that the unfamiliar reader can gain an insight into the motivation, benefits and challenges

involved.

Greater customer expectation for more fuel efficient and higher performing engines that meet increasingly stringent vehicle emissions legislation, has driven automotive manufacturers to invest heavily in developing internal combustion engine technologies. In recent years this trend has been exacerbated by the rising awareness in recent years of energy and environmental issues. In turn, this has led to, for example, tax incentive policies for lower carbon dioxide (CO<sub>2</sub>) emissions – directly proportional to a reduction in fuel consumption – in an attempt to slow global warming (Mundorff *et al.*, 1998).

### 2.9.1 Comparison of Modern Diesel to Spark Ignition Engines

The significant improvement in diesel performance and fuel economy has led to strong demand in Europe for diesel-equipped vehicles. In recent years sharp rises in fuel prices have triggered significant technological development to further improve diesel engines. In 2007, diesel engine-powered vehicles accounted for 50% of the total European market (Herzog *et al.*, 2007). The following summarises some of the advantages and challenges of diesel technology as compared to gasoline and more specifically, spark ignition port fuel injection (PFI) designs. More detail can be found in references such as Stone (1994).

#### Advantages

Major advantages of modern diesel engines over PFI are:

1. Improved fuel economy/maximum efficiency due to:
  - Reduced pumping losses due to the absence of throttling.
  - Reduced frictional losses as diesels in general are designed to operate at lower speeds.
  - The air-fuel mixture is always weak of the stoichiometric air-fuel ratio.
  - Increased compression ratio.
  - Air only being present during the early part of the compression stroke.
2. Improved full load torque due to turbocharging, which is now frequently incorporated in diesel engine designs.



## Challenges

Modern diesel engine challenges comprise:

1. Combustion noise as exhibited by the characteristic ‘diesel knock’, which is caused by too large a rapid combustion period in which self-ignition occurs too slowly. This can be overcome by careful control of the initial fuel injection. However, the more rapid the combustion, the more efficient the engine is, so there is thus a trade-off between fuel economy and minimising combustion noise. Modern diesel engines typically feature an electronic fuel injection system (also known as ‘common rail’ fuel injection equipment (FIE)), which allows a small initial fuel injection, known as pilot injection, before the main injection.
2. Cold starting can pose a serious problem. This can be improved by using a higher compression ratio than optimum for fuel economy or power output. Other aids include excess fuel injection, late fuel injection and electrical heating of air, *e.g.* glowplugs.
3. Hydrocarbon (HC) emissions caused by too much fuel being over-diluted at the periphery of the air-fuel mixture after injection, but before combustion, can also be reduced using pilot injection. Advancing the injection timing also reduces HC emissions, but at the expense of increased Oxides of Nitrogen (more commonly known as  $\text{NO}_x$ ) emissions and combustion noise.
4.  $\text{NO}_x$  emissions strongly depend on combustion temperature, oxygen concentration and combustion duration. Thus, turbocharged engines, which have higher combustion temperatures have higher  $\text{NO}_x$  emissions.  $\text{NO}_x$  can be reduced by increasing the rate of injection, retarding the injection timing and exhaust gas recirculation (EGR). The use of cooled EGR further reduces  $\text{NO}_x$  and means that less EGR can be used, which otherwise may lead to increased particulate and HC emissions.
5. Increased particulate emissions. These originate from the fuel-rich side of the air-fuel mixture in the diffusion-controlled combustion phase. This phase can be shortened by increased swirl, more rapid injection and advanced injection timing. However, earlier injection timing increases the combustion noise.

6. New after-treatment technologies are required. Since diesel engines always operate lean of the stoichiometric air-fuel ratio, lean burn  $\text{NO}_x$ -reducing catalysts are required. The use of a post injection can be provided by a common rail system to supply a controlled quantity of HC as a reducing agent for  $\text{NO}_x$  catalysts. Particulate traps are also used, but typically such filters require temperatures of 550-600 degrees Celsius to become effective or are electrically heated and regenerative.

### 2.9.2 Gasoline Direct Injection Engine Technology

Historically, much of this engine research and development effort was concentrated on enhancing compression ignition direct injection (CIDI), *i.e.* modern diesel engines, and spark ignition port fuel injection (PFI) designs (Heywood, 1988). While the diesel engine has superior fuel economy to the gasoline, it generally exhibits a higher noise level and worse startability, particulate and nitric oxide ( $\text{NO}_x$ ) emissions (Zhao *et al.*, 1999). In the last two decades, much automotive research (Preussner *et al.*, 1998; Sasaki *et al.*, 1998; Yi *et al.*, 2000; Alger *et al.*, 2000; Zhao *et al.*, 1999) has focussed on developing an engine that combines the best of both diesel and gasoline engines, *i.e.* diesel-like fuel economy with the higher specific power of the gasoline engine. A promising candidate for satisfying this goal is the gasoline direct injection (GDI) engine. As its name suggests, this technology involves injecting a fuel spray plume directly into the cylinder. This fuel mixes with inducted air and any residual air/fuel/combustion products and is then ignited by the spark plug.

The following summarises a comparison of GDI versus PFI engine design with more detail to be found in Zhao *et al.* (1999).

#### Advantages

Major advantages of GDI engines over PFI are:

1. Improved fuel economy of approximately 20 - 25% resulting from:
  - Reduced pumping losses during unthrottled, stratified operation.
  - Reduced heat losses to the combustion chamber walls during unthrottled, stratified operation.
  - Lower octane requirements.

- Increased compression ratio.
  - Increased volumetric efficiency.
  - Application of fuel cut-off during vehicle deceleration manoeuvres.
2. Improved transient response from improved air charge measurement accuracy and the absence of intake port fuel films.
  3. An extended EGR tolerance limit, which minimises the use of throttling.
  4. Improved starting due to direct and more precise fuel delivery (Anderson *et al.*, 1996).
  5. Decreased CO<sub>2</sub> and cold-start hydrocarbon emissions.
  6. Enhanced potential for system optimisation.

### Challenges

GDI engine limitations comprise:

1. Difficulties in controlling the stratified charge combustion process over the necessary operating range.
2. Complexity of the control and injection technologies required for imperceptible load or operating mode changes.
3. High rate of formation of injector deposits and ignition fouling.
4. High light-load hydrocarbon (HC) emissions.
5. High light-load NO<sub>x</sub> emissions under part-load stratified-charge operation.
6. Soot formation at high-load operation.
7. Increased particulate emissions.
8. New after-treatment technologies are required (Brogan *et al.*, 1998).
9. Increased fuel system component wear as a function of the increased fuel injection pressures and reduced fuel lubricity.
10. Increased rates of cylinder bore wear.

11. Increased electrical system power requirements.
12. Elevated parasitic losses.

Despite these difficulties, automotive manufacturers have in recent years (Kume *et al.*, 1996) incorporated this technology into new vehicles because the potential advantages of GDI engines cannot be ignored.

### 2.9.3 Engine Calibration

The potential benefits of GDI or modern CIDI engine technology can only be realised using a suitable electronic control system (Zhao *et al.*, 1999). This includes a microprocessor controller, which continuously monitors the current engine state via sensors and then using a computer program or control strategy adjusts the engine state via actuators (Cary, 2003).

A typical control strategy is extremely complex, but in simple terms, the required actuator settings for a given engine operating condition are stored in the microprocessor read-only-memory in the form of look-up tables (Holliday, 1995). For example, diesel engines typically include controls on turbocharger boost pressure, main injection timing, pilot injection timing and quantity, common rail fuel pressure and EGR. A development engineer must provide these data to populate the relevant look-up tables for a given engine application, the process for which is referred to as calibration.

In practice, calibration of any GDI or CIDI modern control system involves the trade-off of several competing objectives. The two case studies included in this research comprise the following calibration trade-offs:

1. The development of modern CIDI engines involves the calibration of common rail FIE to deliver an acceptable trade-off between the competing objectives of the legislated emissions (Yun and Reitz, 2003; Hiroyasu *et al.*, 2004):  $\text{NO}_x$ , particulates, HC, Carbon Monoxide (CO) and  $\text{CO}_2$  (or fuel consumption) as well as combustion noise. As such, this calibration task has been formulated as a six objective optimisation problem. The control variables include:
  - Pilot injection, which reduces combustion noise by slowing the combustion rate and in the process degrades fuel consumption.

- Injection timing - advancing it reduces HC emissions, but at the expense of increased  $\text{NO}_x$  and combustion noise.
  - EGR reduces combustion temperatures and hence  $\text{NO}_x$  emissions, but may lead to increased particulate and HC emissions.
2. On GDI engines, robust cold start performance requires a trade-off between the competing objectives of combustion strength and variation, maximum flare speed (engine speed overshoot immediately after engine start), HC emissions (Zhao *et al.*, 1999; Wiemer *et al.*, 2007) as well as sensitivity to any input variation. Furthermore, engine start is a highly transient process and so these criteria have to be considered over subsequent combustion events. This calibration trade-off problem has been formulated as a ten objective optimisation. Relevant control variables include (Heywood, 1988):
- Spark timing, which can be set to maximise combustion strength and simultaneously minimise combustion variation. However, retarded spark timing reduces HC emissions.
  - Fuel injection quantity, rate, pressure and timing can all affect the combustion metrics as well as flare speed and HC emissions.
  - Throttle setting can be increased to maximise combustion strength, but has a significant effect on flare speed and can easily exceed the maximum level acceptable to the customer.

#### 2.9.4 The Need for Model-Based Approaches

Many of the calibration tasks on modern automotive engines require solution of an optimisation problem, which is often multi-objective in nature and subject to constraints. It is possible to conduct this optimisation *online*, *i.e.* with the optimiser connected directly to the test rig if the objectives are aggregated, and one measurement per search iteration is required. A population-based approach to determining the Pareto-optimal front, as assumed in this research, would require many measurements per search iteration and thus be practically prohibitive. In contrast, a model-based approach offers practical advantages in terms of efficiency, re-use and noise-free or repeatable objective function evaluation. An important requirement of such models to represent objective

functions is that they are adequately, computationally efficient to allow practical optimisation. In other words, the required number of generations of an MOEA can be executed in a reasonable period, *e.g.* hours rather than days. A further advantage of a model-based approach is that it is straightforward to develop sensitivity and curvature functions. Such models can be included directly in the multi-objective optimisation to enable robust calibration trade-offs to be developed without the need for additional test data. Where such models/simulations are not available or direct experimental evaluation is more efficient, online optimisation has been used with some success (Knowles, 2009).

## Chapter 3

# Objective Reduction Methods

### 3.1 Introduction

The primary feature of the proposed practical MOODM process outlined in the previous Chapter (see Section 2.8) is the ability to identify and exploit local harmony for potential objective reduction. This chapter describes how this can be achieved and introduces the main elements. These comprise methods to firstly, partitioning the Pareto-optimal front into clusters of like solutions and subsequently, the use of Principal Components Analysis (PCA) to identify objective dependencies per cluster and potentially discard harmonious objectives. In addition, relevant implementation details are explained.

### 3.2 Partitioning of the Pareto-Optimal Front using Clustering

As stated in Section 2.6.2, objective harmony and conflict can vary across the Pareto-optimal front. In order to discover such local objective dependency, if it exists, it is necessary to partition the Pareto-optimal front into groups of like-solutions. This will allow any local objective harmony to be exploited for local objective reduction. In addition, other studies (Yoshikawa *et al.*, 2007; Müller *et al.*, 2008) suggest that sub-dividing the Pareto-optimal front is useful for visualizing high dimensional Pareto-optimal fronts and grouping similar solutions. Clustering approaches are considered for partitioning the Pareto-optimal front in this research programme.

### 3.3 Clustering

To begin with, it is necessary to define some frequently-used terms when describing clustering approaches. Systems for partitioning or classifying data into groups or clusters are either *supervised* or *unsupervised*. In supervised classification, a number of pre-classified data patterns exist and are used to classify new data. By contrast, unsupervised classification is based solely on the data (Xu and Wunsch, 2005). Clustering is the unsupervised classification of patterns (observations, data items, or feature vectors) into groups (clusters) based on similarity (Jain *et al.*, 1999). Intuitively, the goal of the grouping is that data within a cluster are similar to each other, but dissimilar to data in other groups. Representing the data by a few clusters loses certain fine details, but provides significant simplification - especially in terms of revealing the underlying data structure.

A simple example, depicting the results from a k-Means clustering (Macqueen, 1967) analysis applied to a set of two-dimensional pseudorandom data patterns is shown in Figure 3.1. At this stage, the specifics of the clustering algorithm are unimportant, but it can clearly be seen that the raw data comprises three well defined groups and the algorithm appears to have assigned the data membership correctly.

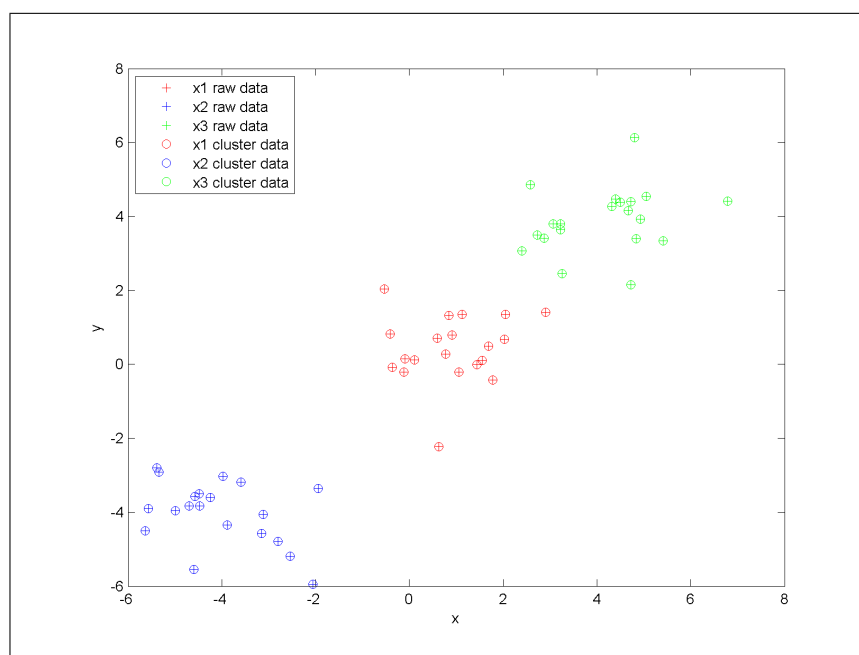


Figure 3.1: Results from a k-Means clustering analysis on a set of two-dimensional pseudorandom data patterns.



By subdividing the Pareto-optimal front objective patterns, as returned by the MOEA, into clusters, it is hoped that any harmony that exists between objective pairs can be identified. If harmony is detected within any cluster, it can be potentially exploited for dimension reduction purposes. The unsupervised nature of the clustering task is therefore important. This is because it is essential that any groupings which result are derived exclusively from properties exhibited by the data patterns. Here the term property implies some form of similarity measure among objective pattern vectors, which has been imposed by the analyst. In other words, data will be grouped based on the appropriate similarity metric.

In the next few sections, the various elements comprising the clustering task are defined and various techniques for them discussed. The intent is not to provide an exhaustive review of clustering, but rather to provide sufficient background to justify the clustering method preferred in this research programme, namely the k\*-Means algorithm (Cheung, 2003).

### 3.3.1 Components of a Clustering Task

Jain and Dubes (1988) state that a typical pattern clustering activity involves the following steps: pattern representation, pattern proximity, clustering or grouping, data abstraction and assessment of output.

The process steps are depicted graphically in Figure 3.2 and described below:

- **Pattern representation.** This refers to the number of classes, the number of available patterns, as well as the number, type and scale of the features available

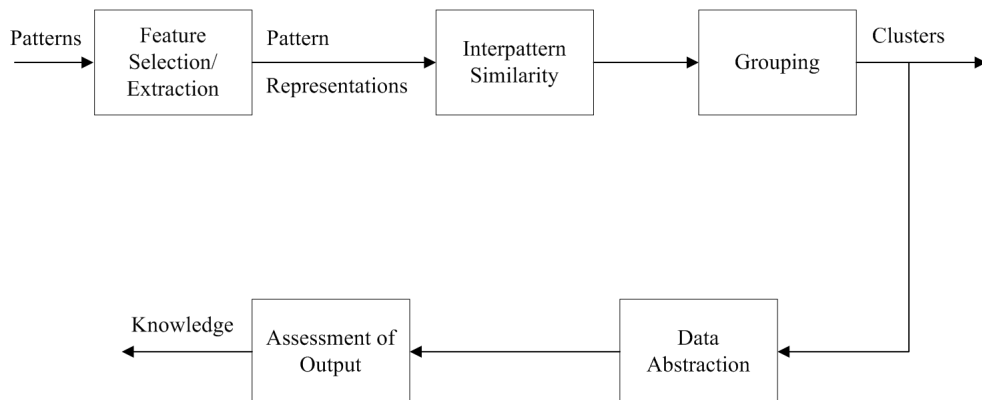


Figure 3.2: Flowchart of a typical clustering process.

to the clustering routine. Feature selection refers to identifying the most effective subset of the original features. Feature extraction refers to the use of one or more transformations of the original features to produce new more significant features.

- **Interpattern Similarity.** Pattern proximity is typically measured by a distance metric defined on pairs of patterns, *e.g.* Euclidean distance.
- **Clustering or grouping of data.** Grouping can be performed using a variety of methods. A broad classification is that the partitioning can be hard, where each pattern is assigned to a single grouping, or fuzzy, where each pattern possesses a variable degree of membership in all identified data groupings.
- **Data abstraction.** This is the process of providing a compact description of the data. It is this aspect that makes clustering such a potentially useful data pre-processing step. Typically the analyst will be interested both in the pattern membership for the cluster (*e.g.* how many data patterns are assigned to the cluster) and descriptive statistics, which illustrate the location, orientation and extent of the cluster.
- **Assessment of output.** This step is concerned with the validity of the clustering. Here questions such as ‘what characterises a *good* or *poor* clustering result?’ arise. After all, every clustering algorithm will produce clusters when presented with data - regardless of whether the data contains obvious groupings or not. Equally, when the data does contain clusters, some algorithms are likely to provide better clusters than others. Clearly, data that does not contain clusters should not be processed by a clustering algorithm. A clustering structure is considered valid if it could not have arisen through chance or as some artificial bi-product of a clustering algorithm.

### 3.3.2 Requirements of a Clustering Algorithm

In order to be effective as part of a MOODM process, a clustering algorithm must fulfill the following requirements:

- **Be efficient to run.** Given that sufficient data density is required to determine the number and location of clusters and that it may need to be run a number of times with different initial settings for the algorithm parameters, it is important

that execution time is relatively short. This is even more relevant if the algorithm is initialised with random settings and consequently needs to be run a number of times to generate reliable results.

- **Generate the correct number of clusters.** Some algorithms assume that the number of clusters is known *a priori*. Since this is unlikely with real world problems, particularly in high dimensions, ideally, the algorithm must define the final number of clusters.
- **Produce a *valid* clustering structure consistent with the downstream process.** Some algorithms make certain assumptions on the distribution of the data to be clustered, *e.g.* the k-Means algorithm (Macqueen, 1967) assumes the raw data is hyper-spherical. However, there is no guarantee that the Pareto-optimal population resulting from the optimisation will conform to a specified distribution. Furthermore, the downstream dimension reduction PCA-based process assumes the clusters are hyper-ellipsoidal.

### 3.3.3 A Review of Clustering Approaches

There follows a brief review of clustering methods with their strengths and weaknesses. More extensive reviews can be found in survey papers such as Jain *et al.* (1999); Berkhin (2002); Xu and Wunsch (2005).

Clustering analysis approaches can be categorised according to the taxonomy as discussed in Jain and Dubes (1988), shown in Figure 3.3 and summarised below. At

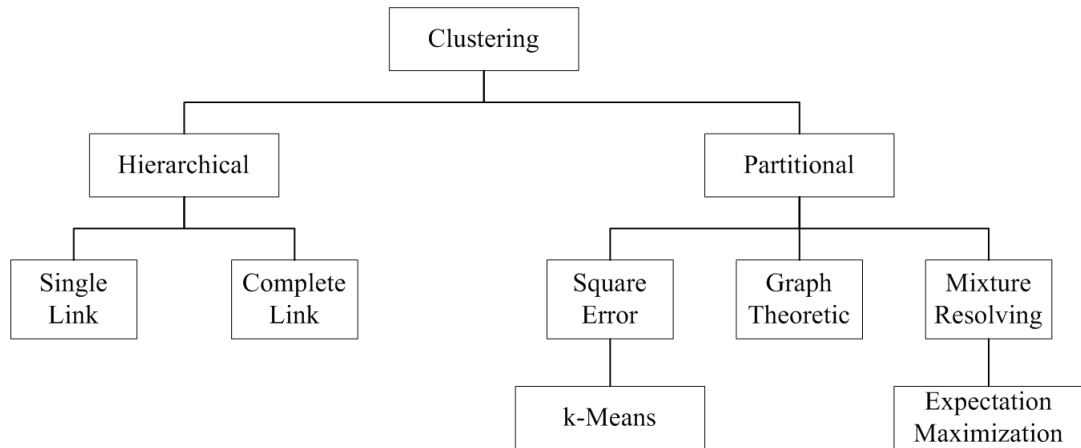


Figure 3.3: A taxonomy of clustering approaches based on Jain and Dubes (1988).

the highest level, the approaches can be split into hierarchical methods, which generate a nested series of partitions, whereas partitional approaches generate a single set of groupings.

### **Hierarchical Clustering**

Hierarchical clustering yields a nested grouping of patterns based on similarity levels and is depicted by a binary tree or dendogram. The Decision Maker (DM) must decide at which level to cut this hierarchy to generate a clustering result. Such representation can be very informative, especially where hierarchical structure in the data exists. However, they are sensitive to noise and outliers, are not capable of correcting previous misclassification and have a high computational cost (Xu and Wunsch, 2005). These weaknesses make such methods unattractive for application to many-objective optimisation problems and will not be considered further.

### **Partitional Clustering**

Partitional clustering algorithms generate a single partition of the data. The generation of all possible solutions to search for the optimal partition according to some criterion is considered practically infeasible (Liu, 1968). Consequently, heuristic algorithms have been developed to generate approximate solutions. For large data sets, such methods have an advantage, where construction of a dendogram would be computationally prohibitive. A weakness of partitional methods is that the number of output clusters must be decided. In an attempt to overcome this, such algorithms are typically run multiple times from different starting states to generate a clustering output, which is ‘best’ in some sense (Jain *et al.*, 1999). The partitional methods relating to the taxonomy in Figure 3.3 are:

**Squared Error Algorithms** are the most popular partitional method, with the k-Means algorithm (Macqueen, 1967) being the most frequently used. From random initial clusters, patterns are reassigned to clusters based on their similarity to the cluster centres until a convergence criterion is achieved. While it is easy to implement and comparatively fast to execute, it is sensitive to the choice of initial clusters and may converge only to a local minimum.

**Graph-Theoretic Approaches** make use of graph theory to describe clustering

problems, where the nodes of a weighted graph correspond to data points and the edges reflect the proximities between pairs of points. A well-known algorithm is based on generating a *minimal spanning tree* (MST) (Zahn, 1971) of the data. The longest MST edges are deleted to produce clusters.

**Mixture-Resolving Algorithms** assume that the data patterns to be clustered belong to one of several distributions. The aim is to identify the parameters of the distributions and a popular approach is *expectation-maximisation* (EM) (McLachlan and Krishnan, 1997). However, EM has a number of disadvantages including sensitivity to selection of initial parameters, convergence to a local optimum and slow convergence (Xu and Wunsch, 2005).

Further to the taxonomy in Figure 3.3, Jain *et al.* (1999) summarise that only the k-Means and Self-Organising Map (SOM) (Kohonen, 1995) algorithms have been tested on large data sets in contrast to artificial neural network, genetic algorithms, tabu search and simulated annealing methods. With the latter approaches it is difficult to adequately calibrate control parameters and they are comparatively, computationally expensive for large data sets. Nevertheless, both k-Means and the SOM require the number of output clusters to be pre-specified.

### 3.3.4 The k\*-Means Clustering Algorithm

The k\*-Means algorithm (Cheung, 2003) is a generalisation of the k-Means algorithm and is designed to address the latter's weaknesses, specifically:

- It allows both spherical and elliptical clusters because it is assumed that the data comprises a mixture of Gaussian densities. This is achieved using Mahalanobis distance,  $l_m$ , as the measure of similarity when clustering data:

$$l_m^2 = (\mathbf{x}_i - \mathbf{x}_c) \mathbf{S}^{-1} (\mathbf{x}_i - \mathbf{x}_c)^T \quad (3.1)$$

where  $\mathbf{x}_i$  is the cluster data,  $\mathbf{x}_c$  is the cluster centre (measure of location) and  $\mathbf{S}$  is the sample covariance (measure of dispersion) of the cluster data.

- The k-Means algorithm suffers from the so-called dead-unit problem (Xu *et al.*, 1993), where some of the cluster centres are initialised comparatively far away from the input data and are never assigned data.

- The number of output clusters must be pre-specified when using the k-Means approach. Importantly, if this number is wrong, an incorrect clustering result will be produced.

The k\*-Means algorithm comprises two main steps. Firstly, a pre-processing procedure is carried out, which assigns at least one cluster centre to each of the initial clusters. The second step involves applying a learning rule to adaptively adjust each centre to a cluster while penalising rival centres. There is no need pre-determine the number of output clusters. Further details are provided in Cheung (2003, pp. 2888).

Since k\*-Means meets the requirements of a clustering algorithm (see Section 3.3.2), it was selected as the algorithm of choice for partitioning the Pareto-optimal front. In addition, a software implementation was readily available to the author (Cary, 2008). To verify that k\*-Means algorithm meets the requirements of clustering algorithm, a number of simulations were run on known data.

### 3.3.5 k\*-Means Simulation Testing

In order to gain some confidence that this algorithm correctly identifies data clusters, simulated data with known correlations, but with no outlying data has been used for testing. Pseudo-random number generators available in Matlab<sup>®</sup> were used to generate the simulation data. Details of the simulations and results are presented in Appendix A.

In summary, the k\*-Means clustering algorithm was able to efficiently and correctly determine the number and location of clusters in multi-variate normal and non-normal data, the latter of which was considered to be significant for Pareto-optimal populations. In other words, these simulation tests provided evidence that the k\*-Means algorithm did meet the requirements for clustering and was a suitable choice for partitioning a Pareto-optimal front.

### 3.3.6 Robust Clustering

Outlying data can have a significant effect on estimation of the mean and covariance of data and hence the generation of Principal Components, which may be attracted to such outlying points. Under these circumstances, Principal Component Analysis (PCA) may be unreliable as a method for dimension reduction (Hubert *et al.*, 2005).

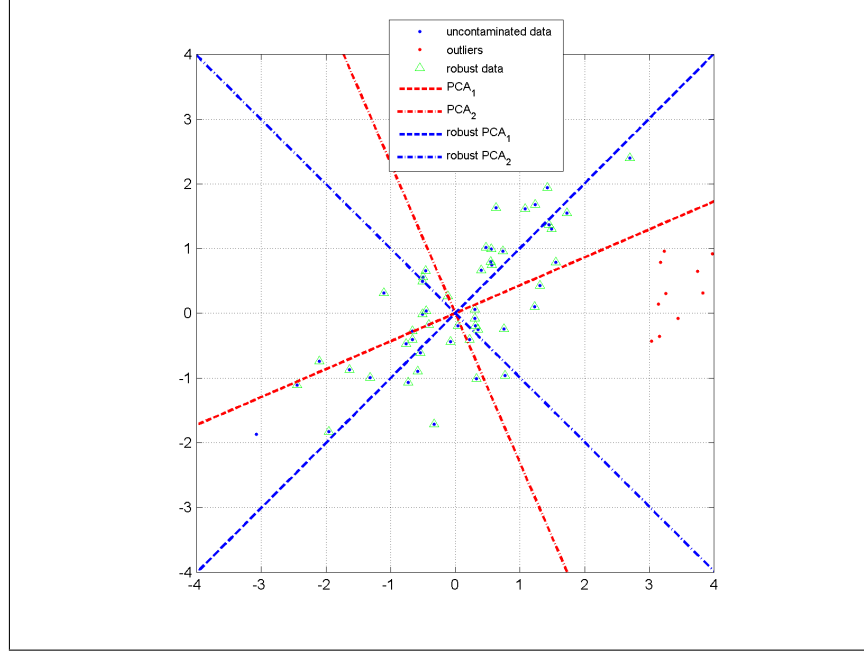


Figure 3.4: Effect of outliers on PCA and robust PCA.

This vulnerability of PCA can be demonstrated with an example, where a number of outliers are added to some ‘uncontaminated’ multi-variate normal random data as shown in Figure 3.4. A PCA is carried out on the covariance matrix of the combined data set and the Principal Axes plotted in red, which are attracted towards the outliers. However, if a robust algorithm is applied to the combined data and a PCA is subsequently performed using the estimated robust covariance matrix, the blue Principal Axes are produced. These align much better with the uncontaminated data, *i.e.* are not influenced significantly by the outliers. This is reinforced when the covariance matrices are compared, where there is good agreement between uncontaminated,  $\mathbf{COV}_u$ , and the robust (resulting from FAST-MCD),  $\mathbf{COV}_r$ , covariance matrices. In contrast, the contaminated covariance matrix,  $\mathbf{COV}_c$ , demonstrates that the presence of outliers can have a significant effect.

$$\mathbf{COV}_u = \begin{bmatrix} 1.3203 & 0.9638 \\ 0.9638 & 1.2090 \end{bmatrix} \quad \mathbf{COV}_r = \begin{bmatrix} 1.1368 & 0.8443 \\ 0.8443 & 1.1419 \end{bmatrix} \quad \mathbf{COV}_c = \begin{bmatrix} 2.6725 & 0.8647 \\ 0.8647 & 1.0448 \end{bmatrix}$$

Mahalanobis distance has historically been used to detect single outliers, where large

Mahalanobis distances suggest potential outliers (Rousseeuw and Van Driessen, 1999). However, if multiple outliers exist Mahalanobis distance is not a reliable approach because:

- It depends directly on the data mean and covariance, which themselves are most affected by the presence of outliers.
- The so-called *masking* effect can arise when small clusters of outliers exist. Removal of one outlier may not significantly influence the covariance estimate if other outliers remain.

Robust estimation of mean and covariance (after clustering, from hereon termed as ‘robust clustering’) are therefore necessary for robust PCA. Another requirement is that the method must exhibit a high breakdown point. The Breakdown point is defined as the smallest proportion of contamination, which has an arbitrarily large effect on estimation of the mean (Rousseeuw and Leroy, 1987). In other words, the approach must be resistant to outliers. For a survey of positive breakdown methods with applications see Rousseeuw (1997). One such method is the minimum covariance determinant (MCD) estimator originally proposed in Rousseeuw (1984, 1985), since much improved in terms of convergence and computational speed and known as the FAST-MCD algorithm (Rousseeuw and Van Driessen, 1999). FAST-MCD gives robust estimation of multivariate location and scatter, *i.e.* robust cluster centres and covariance.

The objective of the algorithm is to find  $h$  observations out of a total of  $n$  whose covariance matrix has the lowest determinant. The MCD estimate of location is the average of these  $h$  points and the MCD estimate of scatter is their covariance matrix. Rousseeuw and Van Driessen (1999) state that the FAST-MCD algorithm is able to deal with sample sizes in the tens of thousands, up to about 100 dimensions, but can only be used when the number of variables,  $nv$  does not exceed the sample size,  $n$  (Hubert *et al.*, 2005). Full details are provided in Rousseeuw and Van Driessen (1999, pp. 13) and the corresponding source code is available from <http://www.agoras.ua.ac.be/Robustn.htm>. However, custom FAST-MCD code (Cary, 2007) has been used throughout this research programme.

In order to determine the robust covariance matrix, a proportion of data,  $h$  is retained. Rousseeuw and Van Driessen (1999) state that a default of  $h = 0.75n$  is a good compromise between breakdown and convergence. For the studies in this research



programme, a subjective value of  $h = 0.95n$  has been used as a compromise between estimating a robust covariance matrix and rejecting data of potential interest, leaning towards retention of the vast majority of data.

### 3.4 Exploiting Linear Associations for Dimension Reduction using PCA

Once the Pareto front has been partitioned, then one approach to identifying linear dependencies between objectives is principal component analysis (PCA). These objective associations are of the same dimension as the number of objectives and require interpretation in order to identify harmonious and conflicting objectives and the strength of this dependency.

### 3.5 Generation of PCA

As described in Section 2.7.1, the motivation for using PCA is to reduce the dimensionality of the many-objective problem, whilst retaining as much as possible of the variation in the Pareto-optimal solutions per objective. This is realised by the transformation of the objectives to a new set of uncorrelated variables that account for the majority of the variation in the original objectives. Such variables are commonly referred to as Principal Components (PCs), the analysis of which is comprehensively described in Jolliffe (2002).

#### 3.5.1 PCA Definition

If  $\mathbf{x}$  is a vector of  $nv$  random variables, the first step of PCA is to find a linear function  $\alpha_1^T \mathbf{x}$  (or PC) of the elements of  $\mathbf{x}$  having maximum variance, where  $\alpha_1$  is a vector of  $nv$  constants  $\alpha_{11}, \alpha_{12}, \dots, \alpha_{1nv}$ , so that,

$$\alpha_1^T \mathbf{x} = \alpha_{11}x_1 + \alpha_{12}x_2 + \dots + \alpha_{1nv}x_{nv} = \sum_{j=1}^{nv} \alpha_{1j}x_j.$$

where  $\alpha_1$  is the vector of coefficients or factor loadings for the  $p$ th PC.

The next stage is to find a linear function  $\alpha_2^T \mathbf{x}$  uncorrelated with  $\alpha_1^T \mathbf{x}$ , but having maximum variance. This process is repeated until at the  $p$ th stage a linear function

$\alpha_k^T \mathbf{x}$  is found having maximum variance and uncorrelated with  $\alpha_1^T \mathbf{x}, \alpha_2^T \mathbf{x}, \dots, \alpha_{k-1}^T \mathbf{x}$ . If there is objective redundancy then most of the variation in the Pareto-optimal solutions per objective will be explained by  $rpc$  PCs where  $rpc \ll nv$ .

### 3.5.2 PCA Properties

The properties of PCs have important implications for dimension reduction as follows:

- The eigenvectors (or principal axes)  $\alpha_p$  are orthogonal to each other and correspond to linearly independent directions in the ellipsoid describing the data under consideration.
- The eigenvalues  $\lambda_p$  measure the amount of variation explained by each PC and are sorted to define successively smaller amounts of variance corresponding to successively shorter principal axes.

As a direct consequence, retention of the largest PCs will explain the majority of the variance and because the PCs are orthogonal, any that are discarded do not affect those retained.

#### PCA using the Correlation Matrix

It is more common to define a vector,  $\mathbf{z}$ , of PCs as

$$\mathbf{z} = \mathbf{A}^T \mathbf{x}^*$$

where  $\mathbf{A}$  now consists of eigenvectors of the correlation matrix and  $\mathbf{x}^*$  consists of standardised variables (having zero mean and unit standard deviation). The main disadvantage of using PCs derived from covariance matrices is that the PCs are sensitive to variables, which have significantly different units of measurement. As this is the case with objective functions typically traded-off in automotive internal combustion engine design, then PCs based on correlation matrices should be used and have been from hereon in this research programme.

#### Zero Eigenvalues

Although it is relatively unusual, it is possible for one or more of the eigenvalues of the correlation matrix to be very close to or equal to zero (Jolliffe, 2002). Any PC with a

zero eigenvalue exhibits a pure linear dependency between the variables, implying that one of them is redundant and can be discarded without losing any information.

### Pairwise Correlations

As described previously, PCs are linear functions of the data (*i.e.* the Pareto-optimal solutions in objective space) defining the principal axes and explaining the variation in the data. These functions are projections of the data onto principal axes, which are of the same dimension and number as the dimension of the data. In other words, if there are six objectives, then there will be six PCs, each of which is six dimensional. By contrast, correlations between objectives can be calculated, but these are pairwise comparisons and unlike PCs, cannot identify dependencies which exist among more than two objectives at a time.

To illustrate this weakness of pairwise correlations, an example is presented in Table 3.1. Initially, a 10 row by 5 column uniformly distributed random data set was created using the Matlab<sup>®</sup> command *rand*, where each column is chosen to represent an objective. To this matrix a sixth column (objective) was added, which equalled the fifth objective divided by 3.5, *i.e.* the Objective 6 has a pure linear dependency or is perfectly correlated with Objective 5. As expected, the resulting correlation matrix identifies this correlation and is shown shaded in Table 3.1. If the Objective 6 is changed to be equal to (Objectives 4 + 5)/2, so that the Objective 6 still has a pure linear dependency, then this is not identified by the correlation matrix as shown in Table 3.1. By contrast, if the PCs are computed (using the Matlab<sup>®</sup> command *pcacov*), then from the resulting sorted eigenvalues,  $\lambda$ , and the matrix of corresponding eigenvectors,  $\mathbf{v}^p$ , displayed in columns for each  $p$  PC as shown below, the last eigenvalue is zero and is evidence of a pure linear dependency in PC6. Examination of the last column, *i.e.* PC6, in  $\mathbf{v}^p$  reveals that only Objectives (rows) 4-6, shown shaded, contribute to this PC and there exists a pure linear dependency or perfect correlation between them.

In summary, PCA has not only been able to detect the pure linear dependency, but also has identified which objectives are affected. It should be noted that the PCA is based on the correlation matrix and therefore the linear dependency is in scaled objectives.

Montgomery and Peck (1991, pp. 318) define a threshold of 1,000 for the ratio of the largest eigenvalue to smaller eigenvalues (also known as *condition number*)

to indicate strong evidence of a linear dependency. For this research programme, a threshold of 10,000 has been used, suggesting very strong evidence.

Table 3.1: Comparison of pairwise correlations for pure linear dependencies.

Data matrix with column 6 = column 5 / 3.5	0.6443	0.2077	0.3111	0.5949	0.0855	0.0244
	0.3786	0.3012	0.9234	0.2622	0.2625	0.0750
	0.8116	0.4709	0.4302	0.6028	0.8010	0.2289
	0.5328	0.2305	0.1848	0.7112	0.0292	0.0083
	0.3507	0.8443	0.9049	0.2217	0.9289	0.2654
	0.9390	0.1948	0.9797	0.1174	0.7303	0.2087
	0.8759	0.2259	0.4389	0.2967	0.4886	0.1396
	0.5502	0.1707	0.1111	0.3188	0.5785	0.1653
	0.6225	0.2277	0.2581	0.4242	0.2373	0.0678
	0.5870	0.4357	0.4087	0.5079	0.4588	0.1311
Correlation matrix corresponding to above data matrix	1	0	0	0	0	0
	-0.4400	1	0	0	0	0
	-0.0601	0.4315	1	0	0	0
	-0.0504	-0.1198	-0.7009	1	0	0
	0.1851	0.6112	0.4798	-0.5442	1	0
	0.1851	0.6112	0.4798	-0.5442	1	1
Data matrix with column 6 = (columns 4 + 5) / 2	0.6443	0.2077	0.3111	0.5949	0.0855	0.3402
	0.3786	0.3012	0.9234	0.2622	0.2625	0.2623
	0.8116	0.4709	0.4302	0.6028	0.8010	0.7019
	0.5328	0.2305	0.1848	0.7112	0.0292	0.3702
	0.3507	0.8443	0.9049	0.2217	0.9289	0.5753
	0.9390	0.1948	0.9797	0.1174	0.7303	0.4238
	0.8759	0.2259	0.4389	0.2967	0.4886	0.3927
	0.5502	0.1707	0.1111	0.3188	0.5785	0.4486
	0.6225	0.2277	0.2581	0.4242	0.2373	0.3307
	0.5870	0.4357	0.4087	0.5079	0.4588	0.4834
Correlation matrix corresponding to above data matrix	1	0	0	0	0	0
	-0.4400	1	0	0	0	0
	-0.0602	0.4315	1	0	0	0
	-0.0504	-0.1199	-0.7009	1	0	0
	0.1850	0.6112	0.4798	-0.5442	1	0
	0.1816	0.6346	0.0430	0.1050	0.7772	1

$$\lambda = \begin{bmatrix} 2.7610 \\ 1.5079 \\ 1.3191 \\ 0.3384 \\ 0.0736 \\ 0.0000 \end{bmatrix}$$

$$\mathbf{v}^p = \begin{bmatrix} -0.0216 & 0.0069 & -0.8473 & -0.3423 & 0.4056 & 0.0000 \\ 0.4728 & -0.2743 & 0.4066 & -0.2113 & 0.7009 & 0.0000 \\ 0.4179 & 0.4743 & 0.0969 & -0.6805 & -0.3577 & 0.0000 \\ -0.3475 & -0.6100 & 0.1048 & -0.5088 & -0.2186 & 0.4354 \\ 0.5621 & -0.0908 & -0.2363 & 0.3403 & -0.1750 & 0.6881 \\ 0.4057 & -0.5652 & -0.2015 & 0.0218 & -0.3714 & -0.5805 \end{bmatrix}$$

### Rules for Principle Component Reduction

The main purpose of using PCA is to reduce the dimensionality of the problem by replacing the variables by a smaller number of PCs which account for most of the variation. Several rules have been developed to identify this smaller number of PCs, some of which are *ad hoc* rules of thumb, but are nevertheless intuitive to use and work in practice and it is these that are described below. Other rules exist based on hypothesis tests, which require distributional assumptions and could be restrictive in nature; there are also statistically-based rules, which can be computationally demanding (Jolliffe, 2002).

**Eigenvalue-1 Criterion** Also known as *Kaiser's rule* (Kaiser, 1960), the principle is to retain and analyse any PC with an eigenvalue of greater than 1. Each variable contributes one unit of variance to the total variance in the data set, because PCA standardises the variables to have a mean of 0 and a variance of 1. Any component that contributes more than one unit is viewed as significant and should be retained, whereas any component that contributes less than one unit is viewed as trivial and is ignored. Advantages of this rule is its simplicity as there is no subjectivity and that it

has been shown to identify the correct number of components for a small to moderate number of variables (Stevens, 1986). However, this rule can lead to retaining the wrong number of PCs where there are a large number of variables. Also, injudicious use may result in a component with an eigenvalue of 0.999 being discarded while one with an eigenvalue of 1.001 is retained.

**Scree Test** This rule, as discussed by Cattell (1966), involves plotting the eigenvalues for each PC to hopefully reveal an elbow or break between the larger, significant values and the smaller, insignificant ones. If there are several breaks, then the last big break should be selected to identify which components to retain. The theory is that the trivial components form the scree at the bottom of a steep rock face. This approach has provided reasonably accurate results provided the sample is large (Stevens, 1986). However, this rule is even more subjective as there can be ambiguity in determining the break, *e.g.* where there are no obvious changes in gradient.

**Proportion of Variance** A PC can be retained if it accounts for a specified proportion of the variance, calculated as follows:

$$tp = \frac{100(l_p)}{\sum_{p=1}^{nv} l_p} = \frac{100(l_p)}{nv}$$

where:

- $tp$  exceeds some cut-off threshold  $tp^*$  of say, 10-15%.
- $l_p$  is the  $p$ th eigenvalue.
- $nv$  is the number of variables.
- The total of all eigenvalues is equal to the number of variables as each variable contributes one unit of variance.

Alternatively, the cumulative percentage of variance can be calculated as below with a typical minimum threshold of 70 to 90%, but for less obvious structures a cut-off of greater than 90% may be required. The number,  $rpc$ , of PCs retained is the smallest value of  $rpc$  whose cumulative percentage of variation exceeds this threshold.

$$tc = \frac{100}{nv} \sum_{p=1}^{rpc} l_p$$

For this research programme the Matlab<sup>®</sup> routine *pcacov* has been used to determine the PC loadings (or eigenvectors), eigenvalues and the percentage of cumulative variation explained by each PC. In turn, *pcacov* uses Singular Value Decomposition (SVD), which is a computationally efficient method for finding PCs.

### 3.5.3 Rotation of Principal Components

In order to simplify the interpretation of PCs, a rotational transformation is often applied to reveal data structure and identify opportunities for dimension reduction. If  $rpc$  PCs account for most of the variation in  $p$ -dimensional data,  $\mathbf{A}_{rpc}$  is the  $(p \times rpc)$  matrix whose  $p$ th column is the eigenvector for the  $p$ th PC, orthogonally rotated PCs have eigenvectors given by the columns of  $\mathbf{B}_{rpc}$ , where  $\mathbf{B}_{rpc} = \mathbf{A}_{rpc}\mathbf{T}$  and  $\mathbf{T}$  is a  $(rpc \times rpc)$  orthogonal matrix.  $\mathbf{T}$  is chosen to optimise one of many possible rotation criteria (Cattell, 1978). It should be noted that the choice of rotation criteria is arbitrary and is often made based on the default option available in the computing software being used. In addition, despite there being several different choices of criteria for orthogonal rotations, as used with this research, there is little difference in the results (Jolliffe, 2002). A popular default and the criterion used in this work is the VARIMAX criterion (Kaiser, 1958). The VARIMAX option is available as an input argument to the Matlab<sup>®</sup> routine *rotatefactors* and has been used throughout this research programme. In the following section, a graphical and numerical example of rotation is described.

### 3.5.4 A Graphical Explanation of PCA

Principal component analysis can be considered to be a rotation of the axes of the original coordinate system to orthogonal axes or Principal Axes, which align with the direction of maximum variation in the data. Thus, a PCA transformation can be interpreted as a rotation of the axes about the origin. Consequently, the distance of each solution from the origin is preserved and therefore objective harmony and conflict relationships are maintained. Using the convention that the principal axes are rotated onto the axes of the original coordinate system, then

$$\mathbf{x}^* = \mathbf{A}\mathbf{z} \quad (3.2)$$

where  $\mathbf{A}$  consists of eigenvectors of the correlation matrix and  $\mathbf{x}^*$  consists of standardised variables.

Graphically, the Principal Axes,  $\varepsilon_x$  and  $\varepsilon_y$  are shown in Figure 3.5, using the convention which is consistent with that used by Matlab<sup>®</sup> command *pcacov* for performing PCA, where a clockwise rotation of the Principal Axes onto the original axes,  $x$  and  $y$ , is defined as negative.

The Principal Axes can be resolved into the unit vector components  $i$  and  $j$  consistent with the orientation of the original axes and described by the direction cosines of the angle  $\theta$ . From Figures 3.6 and 3.7:

$$i = \cos \theta \varepsilon_x + \sin \theta \varepsilon_y \quad (3.3)$$

$$j = \sin \theta \varepsilon_x - \cos \theta \varepsilon_y \quad (3.4)$$

These components can then be assembled in the form described in Equation 3.2 to describe the PCA rotation:

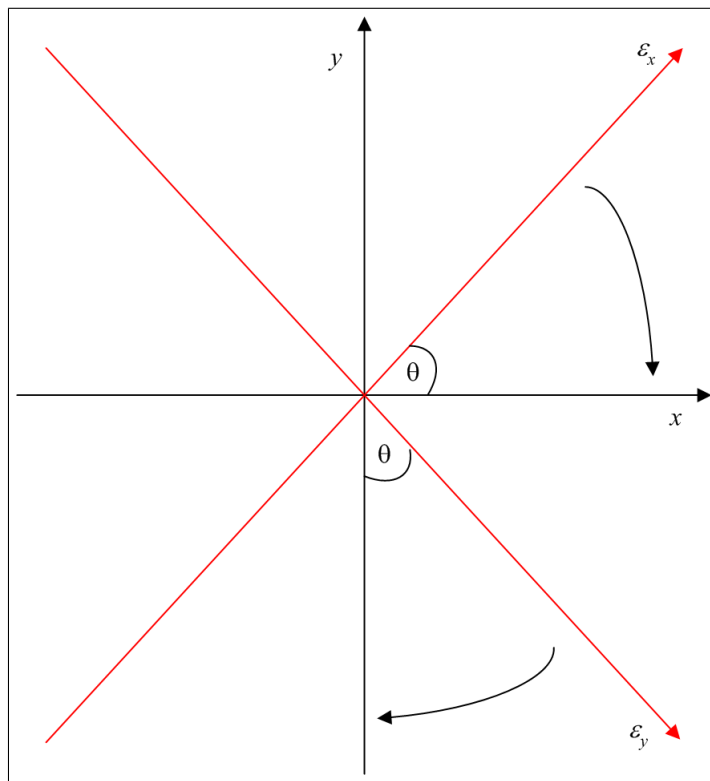


Figure 3.5: PCA rotation convention showing principal axes (red) rotated onto original axes.



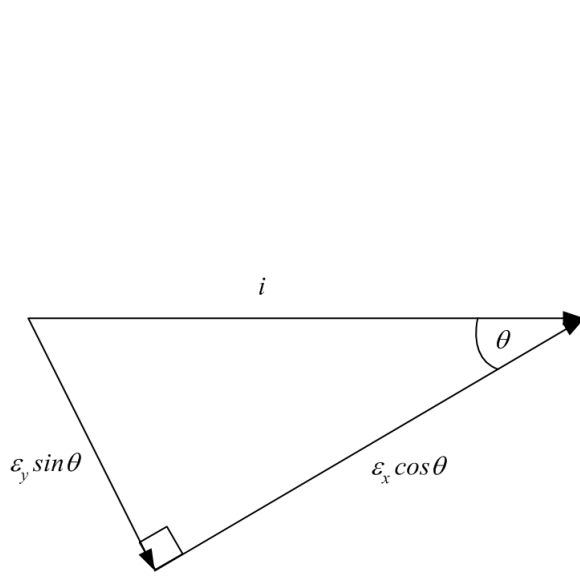


Figure 3.6: Principal axes  $\varepsilon_x$  and  $\varepsilon_y$  resolved into  $i$  direction.

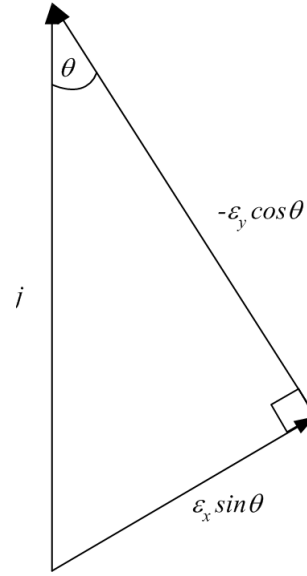


Figure 3.7: Principal axes  $\varepsilon_x$  and  $\varepsilon_y$  resolved into  $j$  direction.

$$\begin{bmatrix} i \\ j \end{bmatrix} = \begin{bmatrix} \cos \theta & \sin \theta \\ \sin \theta & -\cos \theta \end{bmatrix} \begin{bmatrix} \varepsilon_x \\ \varepsilon_y \end{bmatrix} \quad (3.5)$$

Thus, the eigenvector matrix,  $\mathbf{A}$  represents the matrix of direction cosines:

$$\mathbf{A} = \begin{bmatrix} \cos \theta & \sin \theta \\ \sin \theta & -\cos \theta \end{bmatrix} \quad (3.6)$$

As an example, performing PCA on the same single cluster data in two variables from Appendix A using the Matlab<sup>®</sup> command *pcacov* results in:

$$\mathbf{A} = \begin{bmatrix} \cos \theta & \sin \theta \\ \sin \theta & -\cos \theta \end{bmatrix} = \begin{bmatrix} 0.7071 & 0.7071 \\ 0.7071 & -0.7071 \end{bmatrix} \quad (3.7)$$

where,  $\theta = -\frac{\pi}{4} = -45^\circ$ .

A plot of the data and resulting Principal Axes aligned with the maximum variation in the data is shown in Figure 3.8, where  $PC_1$  subtends an angle of  $-45^\circ$  with the  $x_1$  axis.

The benefit of a rotation of the Principal Components as described in Section

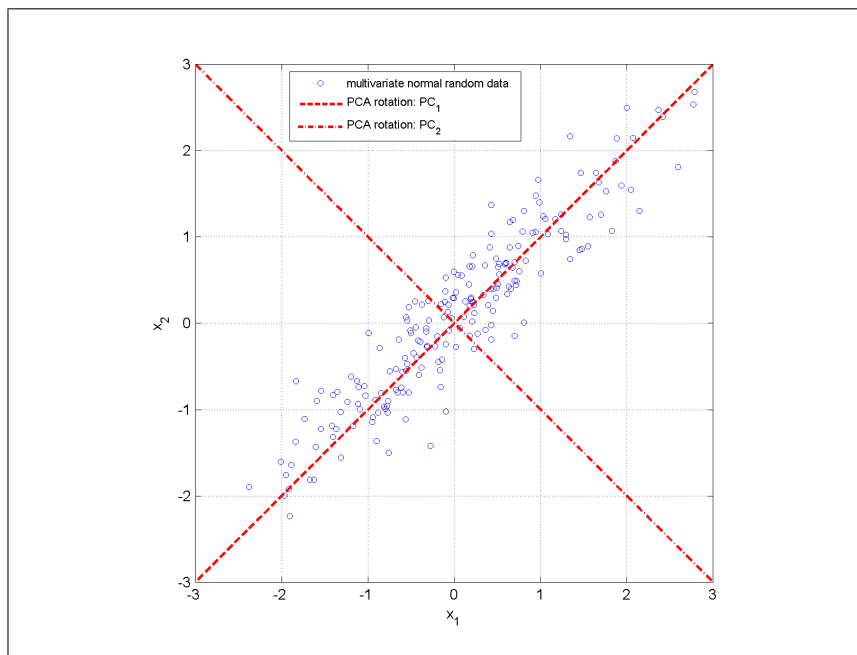


Figure 3.8: PCA rotation on single cluster data in two variables from Appendix A.

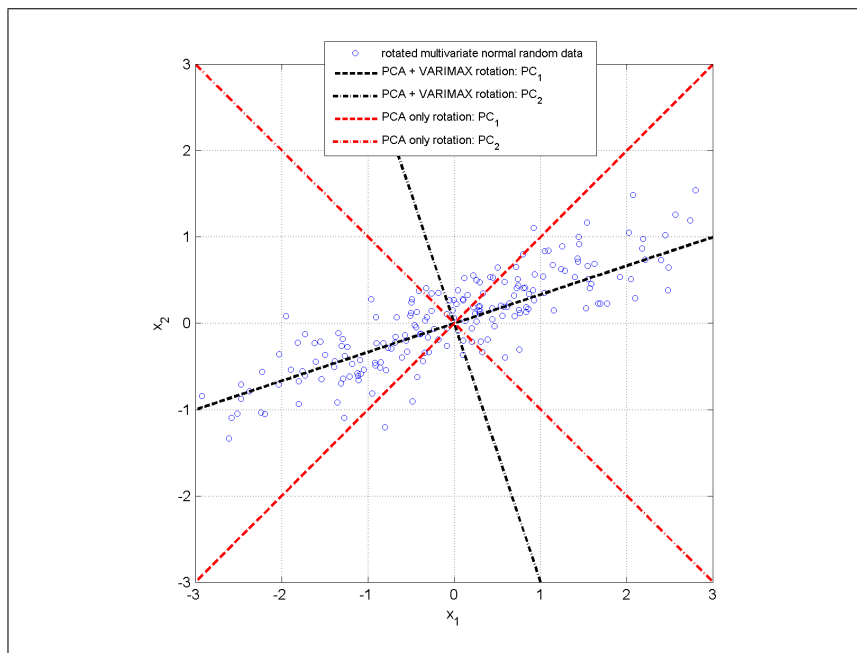


Figure 3.9: PCA + VARIMAX rotation on single cluster data in two variables from Appendix A.

3.5.3 can be demonstrated graphically if the same data is artificially rotated through an angle of  $\frac{\pi}{8}$ , PCA is performed and then an orthogonal rotation is applied. The VARIMAX rotation is one such orthogonal transformation applied to allow the eigenvectors resulting from a PCA to be more easily interpreted so as to reveal data structure. The VARIMAX rotation maximizes the squared eigenvector coefficients (or loadings) in each PC so that the large loadings are increased and the small ones are decreased. The goal is to generate a solution such that each PC has a small number of large negative and positive loadings (or correlations), which may reveal much more clearly evidence of objective harmony and conflict. Graphically, as can be seen in Figure 3.9, the additional orthogonal rotation used here aligns much better with the variation in the data and the corresponding loadings,  $\hat{\mathbf{A}}$  reveal a more discriminating structure than the loadings,  $\mathbf{A}$  from just the PCA rotation as shown below:

$$\mathbf{A} = \begin{bmatrix} 0.7071 & 0.7071 \\ 0.7071 & -0.7071 \end{bmatrix} \quad (3.8)$$

$$\hat{\mathbf{A}} = \begin{bmatrix} -0.9487 & 0.3162 \\ -0.3162 & -0.9487 \end{bmatrix} \quad (3.9)$$

### 3.5.5 Rules for Variable Reduction

With problems comprising many variables (objectives), a subset containing the majority of information can often be found. There have been several studies of various variable selection methods including Jolliffe (1972, 1973). Al-Kandari (1998) concluded that few of the techniques tried were uniformly inferior and none was uniformly superior. Deb and Saxena (2005) have suggested rules which have some correspondence to one method type examined by Jolliffe (2002).

- For the 1st PC select the most positive and negative objectives.
- For subsequent PCs check if the corresponding eigenvalue is  $> 0.1$  or not.
  - If not, choose the objective corresponding to the highest absolute element in the eigenvector.

- If it is, various cases are considered:
  - \* If all the elements of the eigenvector are positive, chose the highest element.
  - \* If all the elements of the eigenvector are negative, chose all the objectives.
  - \* If the value of the highest positive element  $pos$  is less than the absolute value of the most-negative element  $neg$ , two different scenarios are considered:
    - If  $pos \geq 0.9 |neg|$ , choose two objectives corresponding to  $pos$  and  $neg$ .
    - If  $pos < 0.9 |neg|$ , choose only the objective corresponding to  $neg$ .
  - \* Similarly, if  $pos > |neg|$ , two other scenarios are considered:
    - If  $pos \geq 0.8 |neg|$ , choose both objectives corresponding to  $pos$  and  $neg$ .
    - If  $pos < 0.8 |neg|$ , choose only the objective corresponding to  $pos$ .
- Further reduction was considered based on the reduced correlation matrix to investigate if there still exists a set of objectives having identical positive or negative correlation coefficients with other objectives and having a positive correlation among themselves. If so, retain the objective with the largest eigenvalue.

The rules suggested by Deb and Saxena (2005) appear somewhat complicated. They also include a final step based on pairwise correlations, which is inconsistent with using PCA comprising linear functions of, *i.e.* correlations across, all objectives. Alternative rules proposed by the author are simpler and more intuitive. (Objective loading used here is equivalent to eigenvector element used by Deb and Saxena (2005)).

- Use the following threshold for determining if an objective loading magnitude is significant:
  - Loading  $> nobj^{-0.5}$  for  $nobj$  objectives (if the PC axes formed a hyper-sphere then all would contribute equally to the variance in the data).
- If all objective loadings in the same PC have the same sign then they all positively contribute to, *i.e.* increase, the corresponding PC score. In other words, the

objectives in such a PC can be considered as being in harmony. In this case, choose the objective with the largest objective loading magnitude (as it contributes the most to the PC), providing it exceeds  $nobj^{-0.5}$ .

- If the objective loadings in the same PC are in two signed groups, then some of the objectives decrease and some increase the corresponding PC score. In this scenario, the objectives of different signs can be considered to be in conflict. Therefore, choose the largest objective loading magnitude from each group to represent the strongest conflict, providing they exceed  $nobj^{-0.5}$ .

### 3.5.6 Hyper-ellipsoidal Constraint on Principal Components

The k\*-Means algorithm (see Section 3.3.4) clusters the data into ellipse-shaped clusters. From the equations of conic sections, the equation for an ellipse is:

$$\mathbf{A}\mathbf{x}^2 + \mathbf{B}\mathbf{xy} + \mathbf{C}\mathbf{y} = 1 \quad (3.10)$$

In matrix form Equation 3.10 becomes:

$$\begin{pmatrix} X & Y \end{pmatrix} \cdot \begin{pmatrix} A & B/2 \\ B/2 & A \end{pmatrix} \cdot \begin{pmatrix} X \\ Y \end{pmatrix} = 1 \quad (3.11)$$

The k\*-Means algorithm uses Mahalanobis distance as the similarity metric for clustering. When compared to the matrix equation of an ellipse (Equation 3.11), the equation for Mahalanobis distance takes the same form as shown below in Equation 3.12.

$$l_m^2 = (\mathbf{x}_i - \mathbf{x}_c)\mathbf{S}^{-1}(\mathbf{x}_i - \mathbf{x}_c)^T \quad (3.12)$$

where  $\mathbf{x}_i$  is the cluster data,  $\mathbf{x}_c$  is the cluster centre (measure of location) and  $\mathbf{S}$  is the sample covariance (measure of dispersion) of the cluster data.

For multivariate, normally distributed data, Mahalanobis distance approximately follows a Chi-squared distribution,  $\chi_{df}^2$ , with  $df$  degrees of freedom (Rousseeuw and Leroy, 1987). Large robust Mahalanobis distances can be used to detect outliers (as discussed in Section 3.3.6) by comparing them to a quantile of the Chi-squared distribution, *e.g.* the 97.5% quantile (Filzmoser, 2004).

Using  $\chi^2$  as the bound, Equation 3.12 becomes:

$$(\mathbf{X} - \mathbf{X}_c)\mathbf{V}^{-1}(\mathbf{X} - \mathbf{X}_c)^T \leq \chi^2 \quad (3.13)$$

where  $\mathbf{X}_c$  and  $\mathbf{V}$  are the cluster centres and covariance matrix, respectively. Equation 3.13 can be now be used as a hyper-ellipsoidal constraint on a cluster in an optimisation in an attempt to preserve objective correlations within the cluster.

### 3.6 Summary

In order to identify and exploit any incidence of local objective harmony for dimension reduction, methods for partitioning the Pareto-optimal front and consequently determining objective dependencies are described. Following an introduction to clustering, requirements of an algorithm suitable for a MOODM process are defined. A brief review of clustering methods is provided with the k\*-Means algorithm comparing favourably and meeting the aforementioned requirements. Simulation testing is carried out on various known data to verify correct identification of clusters. The FAST-MCD algorithm is described and is a method for dealing with outliers such that they do not have undue influence on the subsequent principal component analysis (PCA). An introduction to PCA follows, which focusses on practical application, together with a graphical explanation. Sets of rules for interpreting PCA results for objective reduction are presented and compared. Finally, a method is described for constraining an optimisation to remain within the cluster boundary in an attempt to preserve objective correlations. This is based on the similarity measure used by the clustering algorithm.

## Chapter 4

# The Many-Objective Optimisation Decision-Making Process

### 4.1 Introduction

In this chapter it is shown how the clustering and principal component analysis (PCA) elements discussed in Chapter 3 can be combined with an efficient search method to produce a systematic dimension reduction process. The goal is to aid the Decision Maker (DM) in discovering opportunities for simplifying the optimisation by reducing the number of objectives at each step. This is achieved by exploiting local harmony, if it exists, in the form of strongly, positively correlated objectives within clusters of the Pareto-optimal population.

The proposed Many-Objective Optimisation Decision-Making (MOODM) process is described with details of the optimisation, clustering and objective reduction elements provided in the following sections. Subsequently, the MOODM process is applied to a real-world automotive problem to demonstrate the principle and the efficacy of the proposed approach.

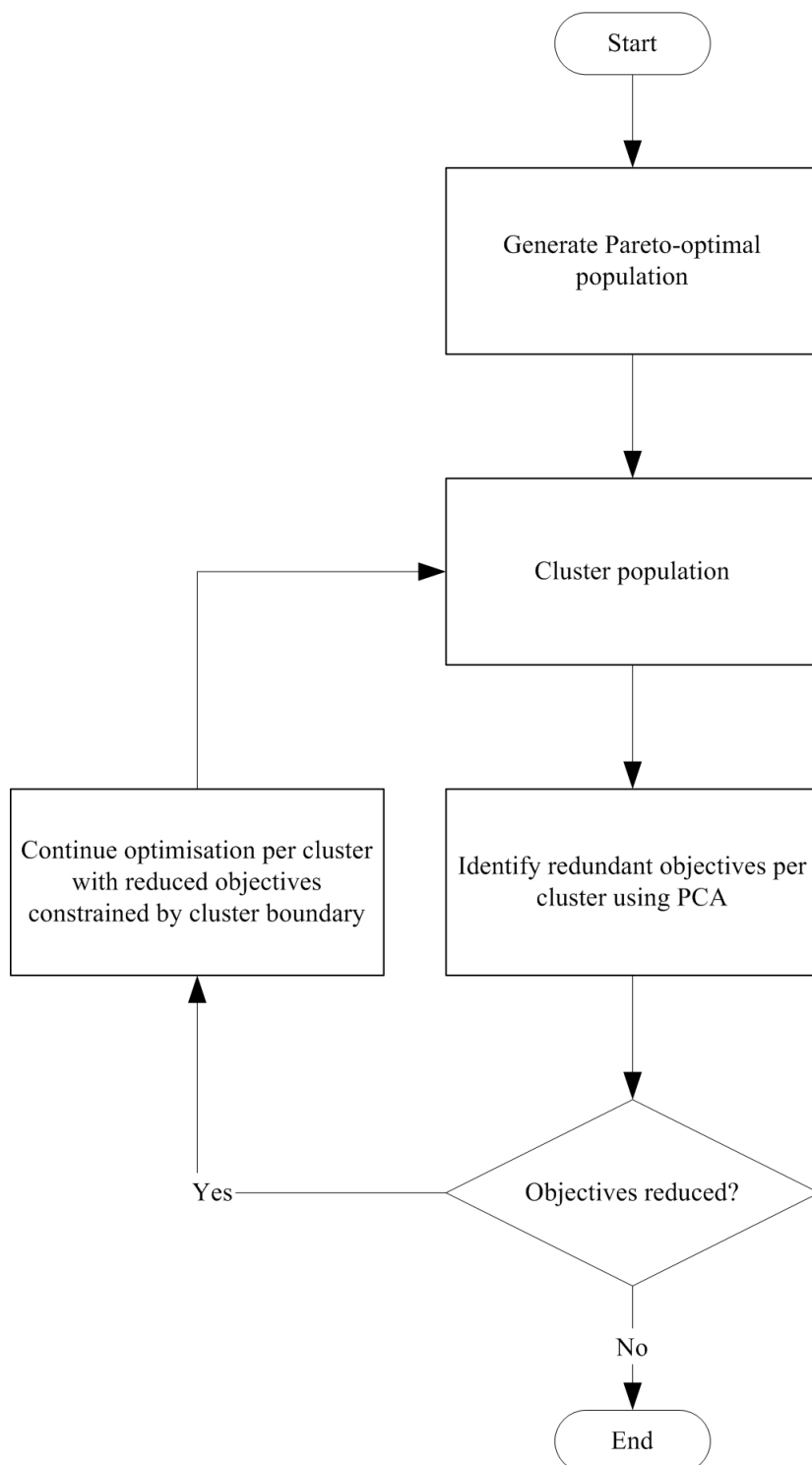


Figure 4.1: Proposed Many-Objective Optimisation Decision-Making (MOODM) Process.



## 4.2 The MOODM Process Description

Figure 4.1 shows the elements of optimisation, clustering and objective reduction via principal component analysis-based heuristic rules combined in a proposed MOODM process. The first step is to efficiently generate a Pareto-Optimal Population (POP). There are many MOEAs that can be used to achieve this, but for many-objective problems, careful consideration must be given to overcome the known issues of lack of search efficiency and very large population sizes required (see Chapter 2). On completion of the optimisation, the resulting POP is partitioned into groups of like-solutions using a clustering algorithm. It is important that the DM is satisfied that there is evidence of the number and location of the clusters and so, cluster verification rules have been developed. Subsequently, for each POP cluster, PCA together with some heuristic rules are applied to reduce objective dimensionality. If no objective reduction is achieved, then the process is terminated. If objective reduction is possible within a cluster, then the process repeats with firstly, optimisation with the reduced objectives subject to the constraint of remaining within the cluster boundary. The process ends when no further objective reduction can be achieved within each cluster.

## 4.3 Generate Pareto-Optimal Population

Consistent with the practical MOODM process described in Section 2.8, the NSGAII algorithm (Deb, Pratap, Agarwal and Meyarivan, 2002) has been modified to incorporate the Progressive Preference Articulation (PPA) method of Fonseca and Fleming (1998a). This allows preferences in the form of objective goals and priorities to be used to direct the search to the DMs region of interest. In addition, the introduction of PPA to NSGAII has been used to mitigate the known search issues of NSGAII in many-objective problems (see Section 2.8). Not only is it a requirement to ensure that the resulting population has converged to the Pareto-optimal front, but also that there is sufficient data density in the population to support the identification of clusters with ‘converged’ correlation matrices. By ‘converged’ it is meant that the iterative calculation of the cluster correlation matrix, generated by the  $k^*$ -Means clustering algorithm, has converged to that of the data in the cluster. To meet these requirements, it is necessary to specify a sufficiently large population and studies such as that described

in Deb (2001, Fig. 276) and the cluster simulation in Section 3.3.5 can be used. For example, for a six objective problem, Deb (2001) suggests a population size of 800 and the cluster simulation on normal data in six variables suggested a population of 1000. Since the computational effort was considered acceptable in the case of the six objective diesel problem, a deliberately large population size of 4000 was chosen, with the view that this could subsequently be reduced if justified.

#### 4.3.1 Apply Objective Goals and Priorities

Preference articulation can be used to direct the optimiser towards the DMs region of interest and so improve search efficiency (Fleming *et al.*, 2005). Furthermore, a progressive approach allows the DM to specify these preferences interactively as information emerges from successive optimisations. The Progressive Preference Articulation method of Fonseca and Fleming (1998a) (PPA<sub>FF</sub>) provides an intuitive and efficient specification of objective goals and priorities (Adra *et al.*, 2007) and is utilised in this research study. However, the description in Fonseca and Fleming (1998a) is relatively compact and so, to aid understanding, a number of simple examples are worked through in Appendix B.

The preference information specified in PPA<sub>FF</sub> approach is in the form of objective priorities and goals. These preferences provide a means of ranking non-dominated solutions and are often naturally available when formulating an optimisation problem. Typically, it is necessary for an analyst to either *prioritise* objectives and/or consider how particular solutions satisfy engineering *goals*. For example, an automotive problem may be to find solutions which are optimal for, in order of priority, fuel economy then combustion stability, subject to meeting a smoke emission constraint. There follows a more formal description.

*Priorities* specify the order of objectives to be optimised according to their importance to the DM, *e.g.* objective A may be more important than objective B, which may be more important than objective C as in the lexicographic approach (Ben-Tal, 1980).

*Goals* define desired values for the objectives and may be:

- a Utopian value for the objective, which is by definition unattainable, but one which the optimiser in question aspires to achieve;

- a target, which the optimiser will try to get as close to as possible;
- a minimum level, which the optimiser must satisfy. This is analogous to satisfying a constraint, where the objective is minimised until the goal is reached.

*Constraints* are often practical limitations in the form of bounds on the decision variable or limits on the allowable domain of an objective function. From Fonseca and Fleming (1998a), constraints can be expressed as inequalities:

$$f(x) \leq g \quad (4.1)$$

where  $f$  is a non-linear function of a decision variable vector  $x$ , and  $g$  is a constant value. Alternatively, the inequality can be strictly less than, or an equality constraint. This inequality applies component-wise, *i.e.* per objective. In other words, the inequality acts as a logical OR relation and is such that there must be at least one objective value in a universe  $\mathcal{U}$  which satisfies all constraints. In practice, this may not be achievable, in which case it may be possible that some of the constraints can be relaxed. These ideas motivate the mathematical concepts of *preferability* and *equivalence*. Preferability/equivalence embodies the concept that a subset of non-dominated solutions may be favoured/equivalent to the remainder, given goals and priorities defined by the DM. For example, consider a simple two-objective, two-individual scenario where both objectives are minimised, as depicted in Figure 4.2. Solutions A and B are both non-dominating, but A satisfies both goals, *i.e.* for A,  $f_1 < g_1$  and  $f_2 < g_2$ , and is therefore preferred. Solution C also satisfies the goals, but A dominates C and is therefore preferred to it. If  $g_1$  and  $g_2$  are constraints that must be satisfied, but once met the objectives are not required to be minimised, then solution A would be equivalent, but not preferable to C. Fonseca and Fleming (1998a) include the definitions of *preferability* and *equivalence* in their definition of a *Comparison Operator*.

#### 4.3.2 The Mathematical Definition of the Comparison Operator

Following from Fonseca and Fleming (1998a), let  $\mathbf{x}$  be some decision variable vector and  $\mathbf{f}$  be a vector function so that  $\mathbf{u} = \mathbf{f}(x_u)$  and  $\mathbf{v} = \mathbf{f}(x_v)$  are two *nobj*-dimensional

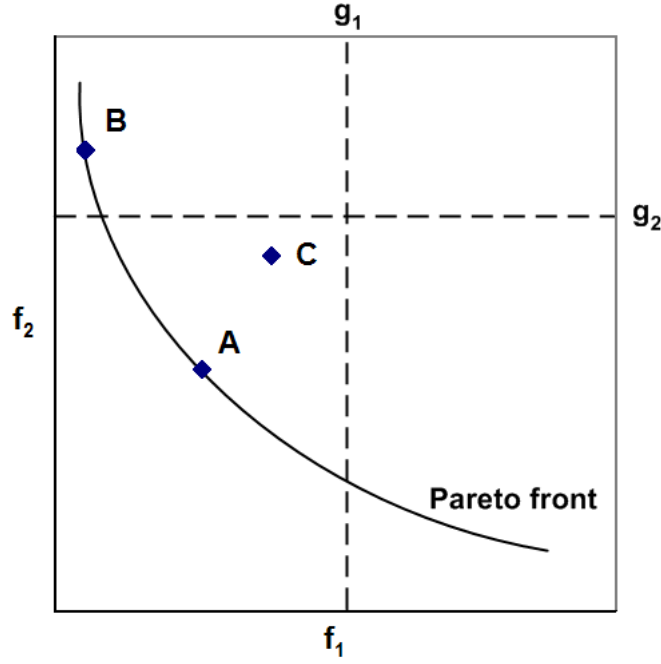


Figure 4.2: Preferability and Equivalence example

objective vectors. Also, let  $\mathbf{g}$  be an  $nobj$ -dimensional preference vector

$$\begin{aligned}\mathbf{g} &= [\mathbf{g}_1, \dots, \mathbf{g}_r] \\ &= [(g_{1,1}, \dots, g_{1,nobj_1}), \dots, (g_{r,1}, \dots, g_{r,nobj_r})],\end{aligned}$$

and,

$$\begin{aligned}\mathbf{u} &= [\mathbf{u}_1, \dots, \mathbf{u}_r] \\ &= [(u_{1,1}, \dots, u_{1,nobj_1}), \dots, (u_{r,1}, \dots, u_{r,nobj_r})];\end{aligned}$$

$\mathbf{v}$  and  $\mathbf{f}$  are defined similarly.

The preference vector  $\mathbf{g}$  can be decomposed into sub-vectors  $\mathbf{g}_i$  where  $i = 1, \dots, r$  associate priorities  $i$  and goals  $g_{i,j_i}$ , where  $j_i = 1, \dots, nobj_i$  to the corresponding objective functions  $f_{i,j_i}$ , components of  $\mathbf{f}_i$ . This convenient permutation of the components of  $\mathbf{f}$  is assumed to be general. The preference vector  $\mathbf{g}$  is defined such that:

- Priority  $i$  is the lowest and  $r$  the highest.

- The number of objectives  $nobj$ , the number of priorities  $r$ , and the number of goals are equal.

To aid understanding, a number of simple examples are worked through in Appendix B.

**Definition 4.1 (Preferability)** Vector  $\mathbf{u} = [\mathbf{u}_1, \dots, \mathbf{u}_r]$  is preferable to  $\mathbf{v} = [\mathbf{v}_1, \dots, \mathbf{v}_r]$  given a preference vector  $\mathbf{g} = [\mathbf{g}_1, \dots, \mathbf{g}_r]$ ,  $(\mathbf{u} \prec_{\mathbf{g}} \mathbf{v})$  iff

$$r = 1 \Rightarrow (\mathbf{u}_r^{\smile} \prec \mathbf{v}_r^{\smile}) \vee \left\{ (\mathbf{u}_r^{\smile} = \mathbf{v}_r^{\smile}) \wedge \left[ (\mathbf{v}_r^{\smile} \not\prec \mathbf{g}_r^{\smile}) \vee (\mathbf{u}_r^{\smile} \prec \mathbf{v}_r^{\smile}) \right] \right\} \quad (4.2)$$

and

$$r > 1 \Rightarrow (\mathbf{u}_r^{\smile} \prec \mathbf{v}_r^{\smile}) \vee \left\{ (\mathbf{u}_r^{\smile} = \mathbf{v}_r^{\smile}) \wedge \left[ (\mathbf{v}_r^{\smile} \not\prec \mathbf{g}_r^{\smile}) \vee (\mathbf{u}_{1,\dots,r-1} \prec_{\mathbf{g}_{1,\dots,r-1}} \mathbf{v}_{1,\dots,r-1}) \right] \right\} \quad (4.3)$$

where:

- The smile,  $\smile$ , and frown,  $\frown$ , annotation denotes the components of  $\mathbf{u}$  which meet and violate, respectively, their corresponding goals.
- $\mathbf{u}_{1,\dots,r-1} = [\mathbf{u}_1, \dots, \mathbf{u}_{r-1}]$  and similarly for  $\mathbf{v}$  and  $\mathbf{g}$ .
- $\mathbf{u} \prec \mathbf{v}$  is derived from the definition of Pareto dominance as given in Fonseca and Fleming (1998a): a given vector  $\mathbf{u} = (u_1, \dots, u_{nobj})$  dominates  $\mathbf{v} = (v_1, \dots, v_{nobj})$  if and only if  $\mathbf{u}$  is partially less than  $\mathbf{v}$ ,  $\mathbf{u} \prec \mathbf{v}$ , that is,

$$\forall i \in \{1, \dots, nobj\}, u_i \leq v_i \quad \wedge \quad \exists i \in \{1, \dots, nobj\} : u_i < v_i.$$

A simple explanation of the Comparison Operator follows from Fonseca and Fleming (1998a). Firstly, vectors  $\mathbf{u}$  and  $\mathbf{v}$  are compared in terms of their objective components with the highest priority, i.e.  $i = r$ , disregarding those in which  $\mathbf{u}_r$  meets the corresponding goals,  $\mathbf{u}_r^{\smile}$ . If both vectors satisfy all goals with this priority in the same way, the next priority down,  $(r - 1)$ , is considered. The process continues until the lowest priority ( $r = 1$ ) is reached, where the priority 1 objective components are compared using Pareto dominance to produce a result.

**Definition 4.2 (Equivalence)** Vector  $\mathbf{u} = [\mathbf{u}_1, \dots, \mathbf{u}_r]$  is equivalent to  $\mathbf{v} = [\mathbf{v}_1, \dots, \mathbf{v}_r]$  given a preference vector  $\mathbf{g} = [\mathbf{g}_1, \dots, \mathbf{g}_r]$ ,  $(\mathbf{u} \equiv_{\mathbf{g}} \mathbf{v})$  iff

$$(\mathbf{u}^{\mathbf{u}} = \mathbf{v}^{\mathbf{u}}) \wedge (\mathbf{u}_1^{\mathbf{u}} = \mathbf{v}_1^{\mathbf{u}}) \wedge (\mathbf{v}_{2,\dots,r}^{\mathbf{u}} \leq \mathbf{g}_{2,\dots,r}^{\mathbf{u}}) \quad (4.4)$$

**Lemma 1** For any two objective vectors  $\mathbf{u}$  and  $\mathbf{v}$ , if  $\mathbf{u} \prec \mathbf{v}$ , then  $\mathbf{u}$  is either preferable or equivalent to  $\mathbf{v}$ , given any preference vector  $\mathbf{g} = [\mathbf{g}_1, \dots, \mathbf{g}_r]$ .

As stated previously, worked examples are detailed in Appendix B, which were found to be very useful in explaining how different parts of the Comparison Operator apply in practice.

### 4.3.3 The Migration of Preferability to NSGAII

The multi-objective optimisation algorithm NSGAII is widely used (Adra *et al.*, 2009) and was already available to the author in Matlab<sup>®</sup> code (MathWorks, 2008b). Consequently, it was decided to modify this NSGAII software to incorporate the Progressive Preference Articulation method of Fonseca and Fleming (1998a) (PPA<sub>FF</sub>). A flowchart representation of the NSGAII Matlab<sup>®</sup> software with the PPA<sub>FF</sub> modifications is shown in Figure 4.3.

These modifications were implemented by The MathWorks Limited at the request of the author and comprise the pseudocode described in Algorithm 1. In the flowchart in Figure 4.3 this is encompassed in the *Rank by Preference* box if the PPA<sub>FF</sub> option is selected. The pseudocode relates to Preferability in Definition 4.1 as follows:

- $obj\_i_{pref} = obj\_i_{fdom} \mid (obj\_i_{feq} \wedge obj\_i_{fpop})$  relates to the first part of Equation 4.3, i.e.  $(\mathbf{u}_r^{\mathbf{u}} \prec \mathbf{v}_r^{\mathbf{u}}) \vee \left\{ (\mathbf{u}_r^{\mathbf{u}} = \mathbf{v}_r^{\mathbf{u}}) \wedge \left[ (\mathbf{v}_r^{\mathbf{u}} \not\leq \mathbf{g}_r^{\mathbf{u}}) \right] \right\}$ .
- $obj\_i_{sdom} = obj\_i_s \prec obj\_i_{pop_s}$  relates to the last part of Equation 4.2, i.e.  $(\mathbf{u}_r^{\mathbf{u}} \prec \mathbf{v}_r^{\mathbf{u}})$ .

There are several points to be made about the subsequent process steps in the flowchart in Figure 4.3:

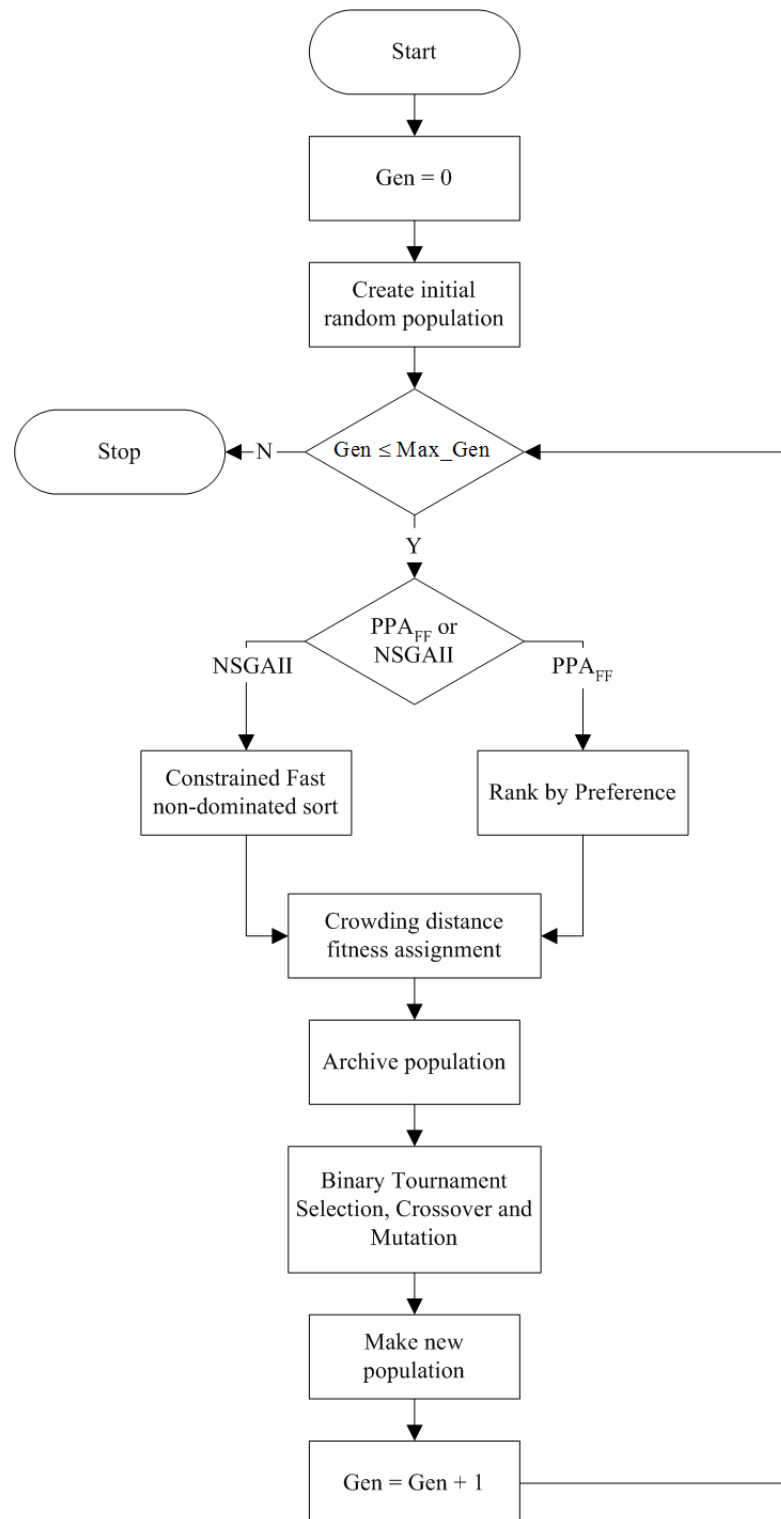


Figure 4.3: NSGAII modified to incorporate the Progressive Progressive Articulation method of Fonseca and Fleming (1998a) (PPA<sub>FF</sub>).

**Algorithm 1** PPA<sub>FF</sub>


---

$obj\_pop_s = obj\_pop \leq goals$	The objective components that meet ('s')
$obj\_pop_f = obj\_pop > goals$	and exceed ('f') the goals respectively.
For each objective individual, $obj\_i$ ,	
For each unique priority from highest to lowest,	
$obj\_i_{fdom} = obj\_i_f \prec obj\_pop_f$	The objective components that exceed the goals and dominate the corresponding objective components in the rest of the population.
$obj\_i_{feq} = obj\_i_f \equiv obj\_pop_f$	The objective components that exceed the goals and equal the corresponding objective components in the rest of the population.
$obj\_i_{fpop} = obj\_pop_s > goals$	The objective components of the population that exceed the goals corresponding to the objective components that meet the goals.
$obj\_i_{pref} = obj\_i_{fdom} \mid (obj\_i_{feq} \wedge obj\_i_{fpop})$	The preferability of $obj\_i$ according to the Comparison Operator.
If any remaining individuals exist and priority level = 1	
$obj\_i_{sdom} = obj\_i_s \prec obj\_pop_s$	The objective components that meet the goals and dominate the corresponding objective components in the rest of the population.
update $obj\_i_{pref}$ matrix	Update preference status of each objective individual.
$rank(obj\_i) = obj\_i_{pref}$	For each objective individual record the individuals preferable to it.
$rank = sum(rank) + 1$	The rank of an objective individual is the number of individuals preferable to it.

---

- In the Crowding distance fitness assignment block: if  $Gen > 0$  AND NSGAI, assign fitness using crowding distance until population filled. If population  $> N$ , choose  $N$  fittest solutions.
- In the Archive population block: the current population is constrained fast non-dominated (CFND) sorted (as with NSGAI), appended to the archive, the archive CFND-sorted and the resulting archive truncated to the same size as the population. Although this use of CFND sorting is inconsistent with the PPA<sub>FF</sub> path, the archive is still a non-dominated population. Resolution of this inconsistency has been left for future work.
- In the Make new population block: if  $Gen > 0$  AND NSGAI,



$\text{New\_Population} = \text{Parent} + \text{Child Populations}.$

## 4.4 Cluster Population

Having generated a Pareto-Optimal Population (POP) using the modified NSGAII, the next stage of the MOODM process (Figure 4.1) is to partition the POP into groups or clusters of like-solutions. As justified in Section 3.3.3, the k\*-Means clustering algorithm is used for this task in this research programme. In the following sections an explanation is provided of the practical aspects of applying this algorithm, namely parameters, usage and cluster verification.

### 4.4.1 k\*-Means Algorithm Parameters

- **Initial number of clusters.** The algorithm assumes that the initial number of clusters is greater than the correct number of clusters. While the correct number of clusters is not known *a priori*, setting this number too high will just waste computational expense. The default value is 10.
- **Learning rate.** This is the rate at which the cluster centres and corresponding covariance matrices are learnt. The default value is 0.001.
- **Convergence tolerance.** This is the difference between successive values of the Maximum Likelihood function minimised by the algorithm. When this difference is less than the convergence tolerance the clustering is deemed to have converged and terminates. The default value is 0.1
- **Maximum iterations.** This is the maximum number of iterations the algorithm runs for before it terminates. The default value is 1000.

All the default values have been recommended from testing by the developer of the k\*-Means Matlab<sup>®</sup> software (Cary, 2008) and were derived from simulation testing.

### 4.4.2 Stochastic Aspects

While the initial number of clusters is specified, the location (or centres) of these clusters is randomly initialised. Subsequently, the cluster centres and covariance matrices are updated iteratively. Just like with MOEAs (Purshouse, 2003), the

stochastic nature of this algorithm justifies running it a number of times to gain confidence that the results are reliable. Consequently, the clustering verification process described in the following section has been designed to ensure consistent and correct results. With many objectives and possible large population sizes, the computational demands may be significant and deserve consideration in deciding how many clustering runs are to be run.

#### 4.4.3 Cluster Verification Rules

Providing the population (resulting from the optimisation) is not too large, it has been found that is efficient and effective to run the clustering several times from different initial numbers of clusters as demonstrated in the following case study. Following the clustering, FAST-MCD (Rousseeuw and Van Driessen, 1999) is used to identify outliers and hence robust Principal Components (see Section 3.3.6); the combination of clustering followed by FAST-MCD is referred to as *robust  $k^*$ -Means clustering*. Finally, the bounds of the data within each cluster are examined to understand if it is justified to keep all clusters. These guidelines have been distilled into a number of Cluster verification Rules: CR1-4 as follows and were applied to the final POP resulting from the multi-objective optimisation. (This POP will be denoted as the *large* POP from this point on).

- CR1** Apply robust  $k^*$ -Means clustering to the large POP from different initial numbers of clusters, *cs*. This should result in the same final number of clusters, *cf*.
- CR2** Randomly sub-sample the data in each cluster resulting from the large POP, assemble into smaller POPs and apply robust  $k^*$ -Means clustering to each. For each of these smaller POPs this should result in the same final number of clusters, *cf*, which should be the same as *cf* from CR1.
- CR3** Compare centres and correlations for each of the smaller POPs. There should be good agreement between corresponding clusters in the different POPs. Select the smallest POP which provides adequate agreement of centres and correlations to the large POP.
- CR4** With the selected POP compare cluster memberships and bounds to justify the

retention of all clusters.

Note: random sub-sampling can be achieved using the Matlab<sup>®</sup> routine *randperm*, which returns a pseudorandom permutation of the input matrix.

## 4.5 Apply PCA to identify Redundant Objectives within each Cluster

The first step in this process is to check the ratio of the largest to smallest eigenvalue (also known as *condition number* (Montgomery and Peck, 1991)) as evidence of a linear dependency. As discussed in Section 3.5.2, if the condition number is sufficiently high, then one of the objectives may be discarded. In such a situation and where objective priorities are specified, the obvious choice would be to discard the lowest priority from the set of objectives in the principal component with a low eigenvalue. Part of the PCA process is to identify those Principal Components (PCs) which explain most of the variance in the cluster data. A simple cumulative percentage of total variation is calculated and a threshold of approximately 95% was used to reject insignificant PCs. The eigenvector coefficients (or factor loadings) of the objectives for each PC are rotated to allow better interpretation and identification of the objectives in harmony and conflict. Heuristic decision-making rules have been developed, which in conjunction with objective priorities, if available, are used to identify redundancy. Finally, it is necessary to check that the retained objectives make sense in terms of expected trade-offs, if known. Such principles have been summarised into a series of Objective reduction Rules: OR1-3 below, which were applied to each cluster after the Cluster verification Rules.

For each cluster:

**OR1** Apply PCA to generate *npc* PCs. Order the resulting eigenvalues,  $\lambda$ , and determine if there is sufficient evidence of linear dependency using the test:  $\lambda_1/\lambda_i \gtrless 10000$ , for  $i = 2 \dots npc$ . For each of the *p*th PCs, which do qualify as near-linear dependencies, examine the corresponding eigenvector and from the coefficients that are not near-zero, select one of the objectives to discard (use the rules OR3a)-c) and/or if objective priorities are specified, select the objective with the lowest priority). Reapply PCA to the remaining objectives to

generate new PCs.

**OR2** Determine the cumulative percentage of total variation for the PCs and retain those  $rpc$  PCs that contribute more than a threshold,  $t_{rpc}$ , where  $t_{rpc} \approx 95\%$ .

**OR3a)** If two or more PCs are retained, rotate the eigenvectors,  $\mathbf{V}^p$  using the VARIMAX transformation (Kaiser, 1958). If only one PC is retained, this represents a line in the objective space and therefore cannot be rotated. Use as a test of significance: eigenvector coefficient,  $v_m^p > m^{-0.5}$  for  $m$  objectives. (If the PC axes formed a hyper-sphere then all would contribute equally to the variance in the data).

**OR3b)** For each PC, if the significant  $v_m^p$ s have the same sign, the corresponding objectives are in harmony. Of these objectives, retain the objective with the significant eigenvector coefficient of greatest magnitude. This rule can be compromised in favour of higher priority objectives.

**OR3c)** For each PC, if the significant  $v_m^p$ s have different signs, the corresponding objectives are in conflict. Of these objectives, retain the objective from each group with the significant eigenvector coefficient of greatest magnitude. This rule can be compromised in favour of higher priority objectives.

## 4.6 Continue Optimisation per Cluster with Reduced Objectives

If objective reduction within a cluster has been achieved, then it is important to preserve the correlations between objectives whilst not overly constraining the subsequent search for improved solutions. This has been achieved by using the cluster hyper-ellipsoidal constraint (see Equation 3.13) developed in the first stage optimisation as an additional constraint in subsequent optimisations.

## 4.7 A Simplified Real-World Example

### 4.7.1 Problem Formulation

This problem concerns calibration of the primary (or ‘base’) actuator settings for a modern diesel engine control system. The objective functions used in this base calibration optimisation case study were built from empirical engine models. These were developed from experimental data collected on a 2.2 litre in-line four cylinder turbocharged common-rail passenger car engine. The data was collected at a single operating condition defined by an engine speed of 1723 RPM, an engine brake torque of 97 Nm, stabilised (fully warm) operating temperatures. The resulting models comprised:

- SFC - corrected specific fuel consumption (g/kWh), which gives a measure of fuel consumption normalised by the engine power output to allow comparison to engines of differing capacity and design (Heywood, 1988).
- Parts - or particulate emissions (g/h).
- NO<sub>x</sub> - or nitrous oxide emissions (g/h).
- HC - or hydrocarbon emissions (g/h).
- CO - or carbon monoxide emissions (g/h). All four of these emissions quantities are required to meet legislated limits and so, could be set as constraints. However, setting these quantities as objectives to be minimised may provide an opportunity to reduce exhaust after-treatment system cost.
- Noise - or combustion noise (dBA) (Brooks *et al.*, 2005). This is formulated as the absolute difference from a noise target, which itself is a function of engine speed and engine torque. An acceptable upper limit for this absolute difference is 3 dBA.

All models had the following inputs, all of which were used as decision variables:

- B - turbocharger boost pressure (hPa).
- M - main injection timing (deg.).

- S - pilot separation ( $\mu\text{s}$ ), *i.e.* the difference in injection timing between the end of the pilot and start of the main injection.
- Q - pilot quantity (mg), quantity of fuel delivered by the pilot injection.
- P - common rail fuel pressure (kPa).
- E - EGR (mg), exhaust gas recirculation.

In order to visualise the results, a smaller three-objective problem was considered, where the objective functions were SFC, Parts and  $\text{NO}_x$ , which were all minimised subject to the model boundary constraint, `model_bdry`. This model boundary is an envelope wrapped around the boundary of the data used to build the models to indicate model extrapolation (MathWorks, 2008a). The model boundary can thus be used as a constraint to ensure that the models are not extrapolated. In other words, this ensures that the model boundary constrains the decision variables to lie within the range of the test data used to build the empirical models (objective functions). The constraint is defined as  $\text{model\_bdry} \leq 0$  and is applied in the modified NSGAI search process in the Archive population block (see Section 4.3.3) where the current population is constrained fast non-dominated (CFND) sorted such that the search is biased against extrapolated or infeasible solutions.

The NSGAI algorithm modified to incorporate preference articulation of Fonseca and Fleming (1998a) ( $\text{PPA}_{\text{FF}}$ ) was used with the following options and parameters specified:

- Number of generations: 2000.
- Population size: 2000 - both of these first two parameters were chosen to provide a reasonable computational effort, *i.e.* it took less than an hour to run; in addition, while a population size of at least 500 was required based on findings from the clustering simulation in Section 3.3.5, the computational expense of such a population size was considered acceptable to be consistent with Cluster verification Rule CR2.
- External archive used and updated every generation (Laumanns *et al.*, 2002).
- Selection operator: tournament of size 2.

- Crossover operator: SBX (Deb and Agrawal, 1995) with probability: 0.7 and distribution index: 20
- Mutation operator: polynomial (Deb and Goyal, 1996) with probability: 0.17 and distribution index: 20 (Deb, Thiele, Laumanns and Zitzler, 2002; Khare *et al.*, 2003).
- Minimum and maximum range limits on the decision variables are listed in Table 4.1.
- Initially all the objectives were minimised subject to the constraint, which corresponds to the *constrained minimisation* formulation in the PPA<sub>FF</sub> method of Fonseca and Fleming (1998a). Despite model\_bdry being defined as a constraint in the engineering context of formulating the optimisation problem, the PPA<sub>FF</sub> approach treats the constraint as the highest priority objective, which must be satisfied first before minimisation of the lower priority objectives. The resulting initial goals and priorities are shown in Table 4.2, where the higher the number, the higher the priority.

Table 4.1: Decision variable ranges.

Decision Variable	Units	Ranges	
		Minimum	Maximum
B	hPa	1014	1035
M	deg.	-3.42	2.57
S	$\mu s$	5100	7140
Q	mg	0.6	3
P	kPa	246293	374551
E	mg	292	503

#### 4.7.2 Three-Objective Diesel Problem Results

##### Clustering and Verification

The resulting Pareto-optimal front is displayed in Figure 4.4. As shown, the Pareto-optimal solutions have been robustly clustered into two groups. For the

Table 4.2: Initial goals and priorities for the three-objective diesel problem, where the last row represents the constraint.

Objective	Goal	Priority
SFC	$-\infty$	1
$\text{NO}_x$	$-\infty$	1
Parts	$-\infty$	1
model_bdry	0	2

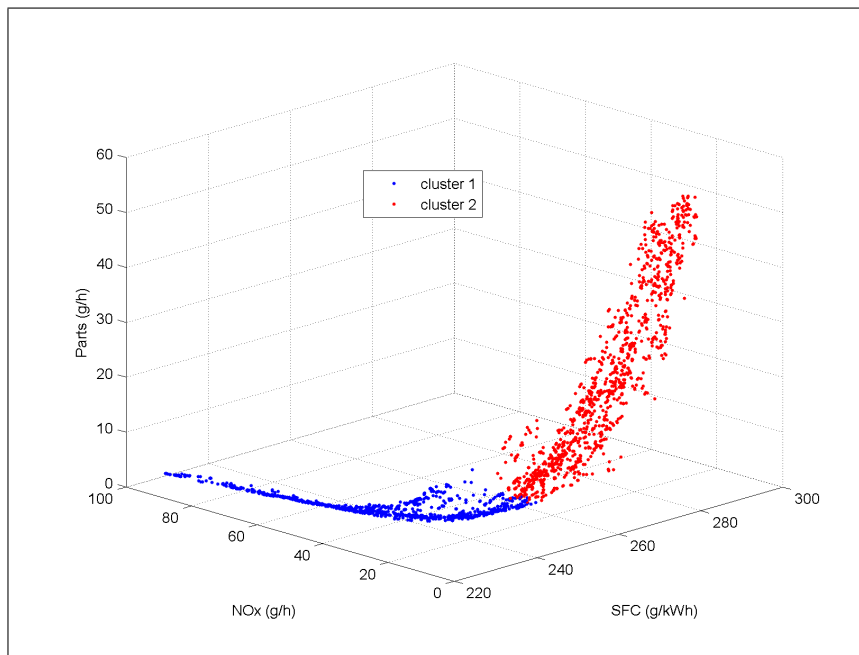


Figure 4.4: 3D Pareto-optimal front from 3-objective diesel problem split into clusters.

clustering of the Pareto-Optimal Population (POP) resulting from the optimisation, the  $k^*$ -Means parameters were set to their default values. The exception was the Maximum Iterations, which was set to 2000 in an effort to ensure the algorithm had sufficient iterations to converge with the larger populations specified here compared to those used previously in simulation testing.

The clustering into like-solutions can be more easily identified from pairwise plots of the objectives as shown in Figure 4.5. Cluster 1 has low diversity in SFC and Parts but high diversity in  $\text{NO}_x$ , whereas Cluster 2 has the opposite - low diversity in  $\text{NO}_x$ , but high diversity in SFC and Parts. Obviously, with such a relatively low dimensional problem and utilising such scatter plots, it can be straightforward to



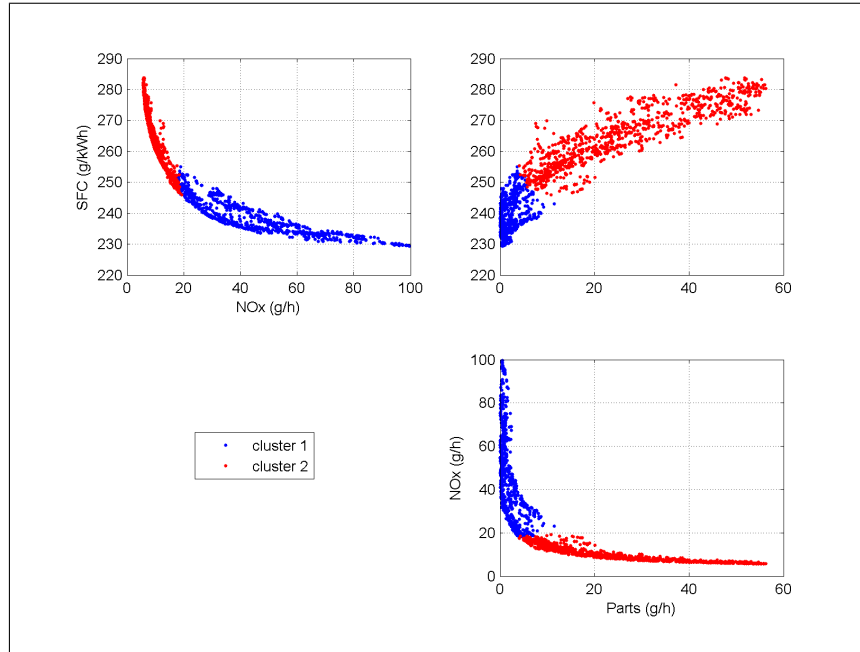


Figure 4.5: Pairwise plot of Pareto-optimal front from 3-objective diesel problem showing clustering of like-solutions.

visualise and hence verify that the clustering has been successful both in terms of the location and number of clusters. However, for higher-dimensional problems the scatter plot approach to visualising the Pareto-Optimal front is more difficult to interpret for the DM and alternative visualisation methods have been used (Miettinen, 1999; Deb, 2001). One such approach is that of Parallel Coordinates Plots (Inselberg, 1985).

Parallel coordinates plots of the two clusters are shown in Figures 4.6 and 4.7. Two plots are required for the two different objective orders possible. Just as with scatter plots, it is possible to see evidence of the two clusters and such plots can be helpful to indicate the number and location of clusters, but can require some interpretation, particularly where the clusters overlap. As the use of scatter and parallel coordinates plots demonstrate, it is possible to verify the number and location of clusters, but it would be altogether more difficult to determine the number and location of clusters without any *a priori* knowledge. This is where algorithms such as k\*-Means (Cheung, 2003), which automatically determine the number and location of clusters are very useful. The Cluster verification Rules from Section 4.4.3 were applied to verify the number and location of clusters of like Pareto-Optimal solutions, the conclusions from

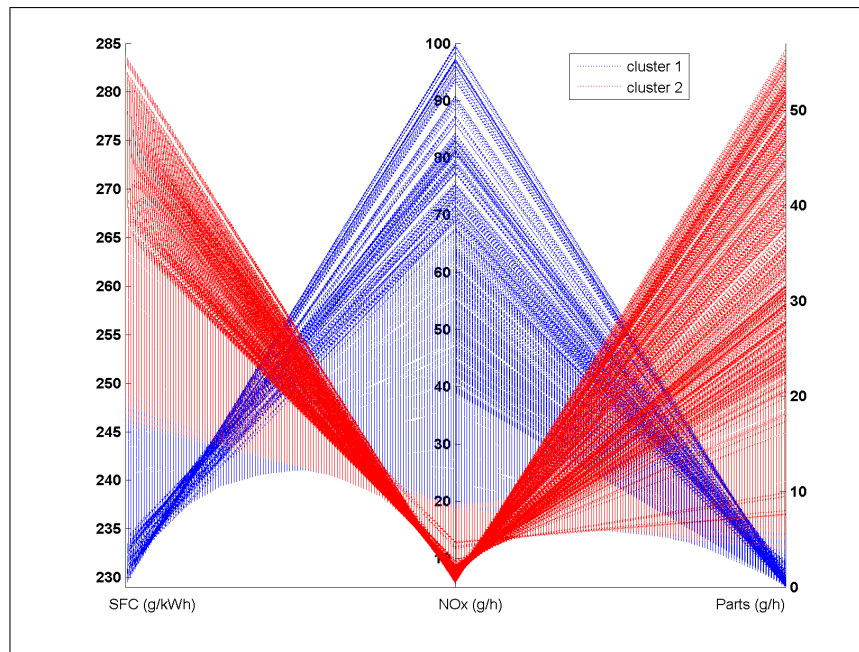


Figure 4.6: Parallel Coordinates plot of Clusters 1 and 2 for objective order: SFC, NO<sub>x</sub>, Parts.

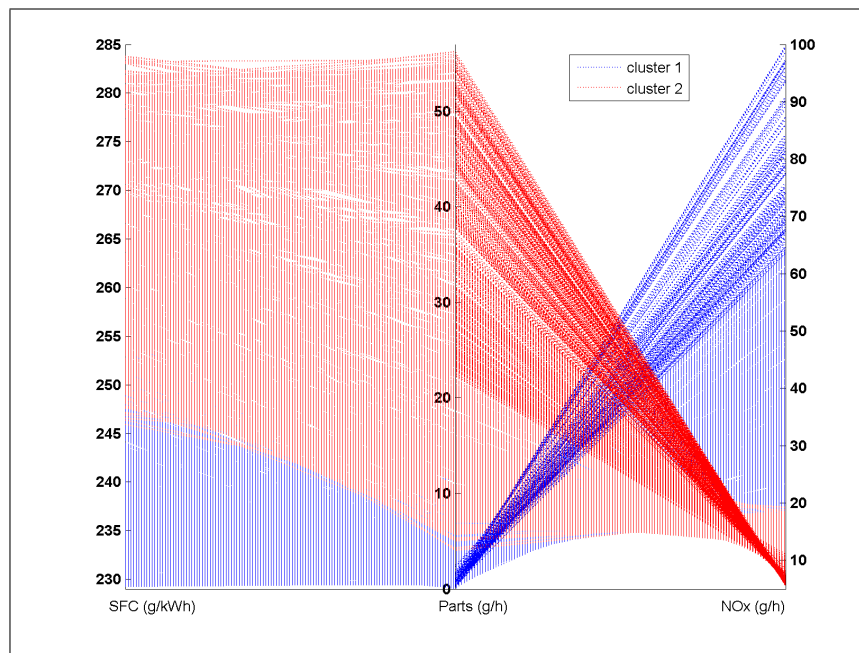


Figure 4.7: Parallel Coordinates plot of Clusters 1 and 2 for objective order: SFC, Parts, NO<sub>x</sub>.

which are listed below. The clustering has been repeated from various initial numbers of clusters and on different size Pareto-optimal population sizes with the results shown in Table 4.3.

Table 4.3: Comparison of clusters resulting from the POP of 2000 with those of randomly sub-sampled POPs.

Cluster Property	Population	Cluster 1			Cluster 2		
Centres	2000	239.5	42.3	2.0	265.2	9.7	26.5
	1000	239.8	41.5	2.0	264.2	10.1	25.7
	500	239.5	43.7	1.9	266.1	9.4	27.7
	250	240.6	39.4	2.2	265.2	9.8	26.0
	100	241.2	38.5	2.5	264.8	9.7	26.5
Correlation Matrices	2000	1			1		
		-0.86	1		-0.95	1	
		0.41	-0.65	1	0.91	-0.90	1
	1000	1			1		
		-0.86	1		-0.95	1	
		0.43	-0.65	1	0.91	-0.90	1
	500	1			1		
		-0.87	1		-0.94	1	
		0.43	-0.60	1	0.91	-0.92	1
	250	1			1		
		-0.88	1		-0.94	1	
		0.51	-0.65	1	0.89	-0.92	1
	100	1			1		
		-0.91	1		-0.96	1	
		0.60	-0.67	1	0.86	-0.89	1

- Applying CR1, robust clustering was repeated on the same population of 2000 Pareto-optimal solutions from  $cs$ : 4, 5 three times, 6 and 8. All runs resulted in a  $cf = 2$ .
- Applying CR2, one of the pairs of clusters (the second run from a  $cs = 5$ ) resulting from the application of CR1 to the POP of 2000, was randomly sub-sampled using the Matlab<sup>®</sup> routine *randperm* into smaller POPs of sizes:

1000, 500, 250 and 100. Robust clustering (retaining 95% of the data) was run on all of the POPs and all yielded  $cf = 2$ .

- Applying CR3,
  - The cluster centres for Objectives 1-3 were determined and in comparison were broadly similar for all POPs, as can be seen in Table 4.3. However, closer examination of Cluster 1 reveals that some divergence in centres is evident with the POPs of 250 and 100 compared to that of 2000.
  - Since the PCA was based on the correlation matrices (in lower triangular form), these were calculated for all POPs and had broadly similar magnitude and structure, as can be seen in Table 4.3. Again, further inspection of the smaller POPs shows some evidence of divergence from the correlation matrices of the POP of 2000.

Consequently, this divergence in centres and correlations justified the selection of the POP of 500 as the smallest POP providing adequate agreement with the POP of 2000.

- Applying CR4 to the selected POP of 500, Cluster 1 had 232 members while Cluster 2 had 268, so both had significant membership. In order to compare objective ranges in the POP of 500, the upper and lower bounds were determined. In terms of minimising objectives with a lower and ideally smaller, range:
  - Cluster 1 was better for Objectives 1 (SFC) and 3 (Parts).
  - Cluster 2 was better for Objective 2 ( $\text{NO}_x$ ).

Therefore both clusters were retained. These results are shown in Table 4.4.

### Principal Component Analysis and potential Objective Reduction

In order to establish any opportunity for objective reduction, the objective reduction rules previously defined in Section 4.5 was followed for each cluster, the results for which are displayed in Table 4.4.

Table 4.4: Clustering and PCA results for the selected POP. For each cluster, the eigenvalues are listed in Principal Component order (largest first), followed by the corresponding cumulative percentage of total variation. Also, the eigenvectors and as appropriate, rotated eigenvectors, are listed by objective for the retained PCs with the selected significant eigenvector coefficients shown shaded.

		Cluster 1			Cluster 2		
Cluster Membership		232			268		
Cluster Bounds	Upper	255.4	99.6	8.3	283.4	18.1	56.2
	Lower	229.3	17.5	0.1	247.0	5.7	5.4
PC Eigenvalues ( $\lambda_p$ )		2.2894			2.8480		
		0.6077			0.0967		
		0.1030			0.0553		
Cumulative % of Total Variation		76.3			94.9		
		96.6			98.2		
		100			100		
PCs retained		PC1		PC2	PC1		
Eigenvectors ( $\mathbf{v}^p$ )	Obj1	-0.5957		-0.4902	-0.5783		
	Obj2	0.6332		0.2008	0.5805		
	Obj3	-0.4942		0.8482	-0.5732		
Rotated Eigenvectors ( $\mathbf{v}^{p'}$ )	Obj1	-0.7599		-0.1322	n/a		
	Obj2	0.6500		-0.1372			
	Obj3	-0.0122		0.9815			

#### Objective Reduction Rules applied to Cluster 1

1. Applying OR1,  $\lambda_1/\lambda_3 = 22.2$ , so there is no evidence of a near-linear dependency.
2. Applying OR2,  $t_2 \approx 95\%$ , *i.e.* retain the first two PCs, which account for approximately 95% of the cumulative percentage total variation.
3. Applying OR3a), the eigenvectors for the two retained PCs were rotated and the threshold for the test of significance is  $3^{-0.5} = 0.5774$ .
4. Applying OR3c) to PC1, the rotated eigenvector coefficients (shaded in Table 4.3),  $v_1^1$  and  $v_2^1$  exceed 0.5774 in magnitude and thus are significant. They have opposite signs and so are in conflict. Therefore, both Objectives 1 (SFC) and 2 ( $\text{NO}_x$ ) are retained.

5. Applying OR3b) to PC2, only the rotated eigenvector coefficient,  $v_3^2$  is significant and so Objective 3 (Parts) is also retained.

Thus, there was no opportunity for objective reduction in Cluster 1. It is useful to refer back to Figures 4.4 and 4.5 to verify these conclusions with regard to Cluster 1. It can be seen that conflict is present between all objectives in the region where  $\text{NO}_x$  is low, justifying their retention in Cluster 1. However, if all of Cluster 1 is considered, it can be argued that the conflict between Objectives 1 (SFC) and 2 ( $\text{NO}_x$ ) is stronger than between 2 ( $\text{NO}_x$ ) and 3 (Parts). This is supported by the fact that the majority (76%) of the cumulative percentage of total variation is accounted for by the first PC, and (applying OR3c) to the unrotated eigenvector) only Objectives 1 (SFC) and 2 ( $\text{NO}_x$ ) would have been significant and retained.

### Objective Reduction Rules applied to Cluster 2

1. Applying OR1,  $\lambda_1/\lambda_3 = 51.5$ , so there is no evidence of a near-linear dependency.
2. Applying OR2,  $t_1 \approx 95\%$ , *i.e.* retain the first PC only. Note this means that the PC represents a line in objective space and cannot be rotated.
3. Applying OR3c) to PC1, all three objectives were similar to the threshold (0.5774) for the test of significance and Objectives 2 ( $\text{NO}_x$ ) and 3 (Parts) were retained (in preference to Objectives 1 and 2) as there are legislated limits on these quantities.

Again, it was useful to refer back to Figures 4.4 and 4.5 to verify these conclusions with regard to Cluster 2. It can be seen that there is evidence of harmony between Objectives 1 (SFC) and 3 (Parts) as supported by the fact that the eigenvector coefficients for these objectives have the same sign. In addition, it could be seen that conflict exists between Objectives 2 ( $\text{NO}_x$ ) and 3 (Parts).

**Further optimisation within Cluster 2** In the previous section the number of objectives in Cluster 2 was reduced from three to two. Therefore, further optimisation was carried out on the two remaining objectives of  $\text{NO}_x$  and Parts. The final population resulting from the original optimisation was used as the initial population

for the further optimisation. In addition, a hyper-ellipsoidal cluster bound constraint, `cluster_bdry`, was determined (see Section 4.6) in an attempt to ensure that the objective search space would remain within Cluster 2, and the corresponding objective correlations, *i.e.* harmony and conflict, would be preserved. The problem formulation used was the same as in Section 4.7.1, except that the maximum number of generations was reduced to 500 to reflect the reduced number of objectives and yet still provide a reasonable computational effort. The corresponding goals and priorities are listed in Table 4.5 and show the additional `cluster_bdry` constraint, which like the `model_bdry` constraint has the same highest priority and must be satisfied first before minimisation of the lower priority  $\text{NO}_x$  and Parts Objectives, *i.e.* a constrained minimisation formulation.

Table 4.5: Goals and priorities for the reduced, two-objective diesel problem.

Objective	Goal	Priority
SFC	$-\infty$	0
Parts	$-\infty$	1
$\text{NO}_x$	$-\infty$	1
<code>model_bdry</code>	0	2
<code>cluster_bdry</code>	0	2

It should be noted that for the discarded Objective, SFC the priority was set to 0, which means:

- that it was ignored, *i.e.* not included in the modified NSGAIIs determination of objective fitness, and
- it was required to determine the ellipsoidal cluster boundary constraint, since this was defined in the original three-objective space and any lower dimensional projection would be give an incorrect boundary.

Although it is useful to calculate the value of the discarded objective to ensure its value has not been degraded with further optimisation of the other two objectives, its inclusion means that the computational burden has not been reduced with further optimisation. The resulting population is plotted against the original Cluster 2 in

parallel coordinates format in Figure 4.8 and in scatter plot format in Figure 4.9. The parallel coordinates plot shows that while it is possible to select a solution that has a lower Parts value, this is at the expense of  $\text{NO}_x$  and vice versa, providing evidence of significant conflict. For the discarded Objective, SFC, the range of solutions has deteriorated somewhat as allowed by the ellipsoidal cluster constraint, but not significantly. It is slightly clearer in this three-objective case to examine the scatter plot, which shows that while the Pareto-optimal front between the preferred and retained Objectives  $\text{NO}_x$  and Parts has improved with the Stage 2 optimisation, this has been at the expense of the discarded Objective, SFC.

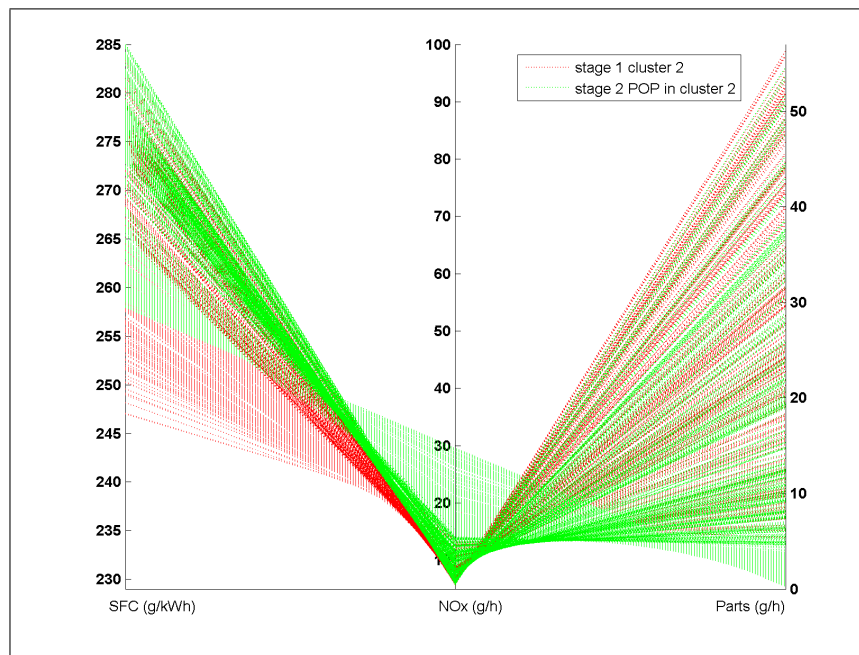


Figure 4.8: Parallel Coordinates plot of Cluster 2 and the Stage 2 POP resulting from the (reduced) two-objective optimisation of  $\text{NO}_x$  and Parts in Cluster 2.

It is important to note that while the ellipsoidal constraint has allowed improved Pareto-optimal solutions with regard to  $\text{NO}_x$  and Parts to be found, it has maintained the same objective correlations as in Stage 1. That is, SFC and Parts are positively correlated or are in harmony, whereas the other Objective pairs of SFC/ $\text{NO}_x$  and  $\text{NO}_x$ /Parts are negatively correlated or in conflict.



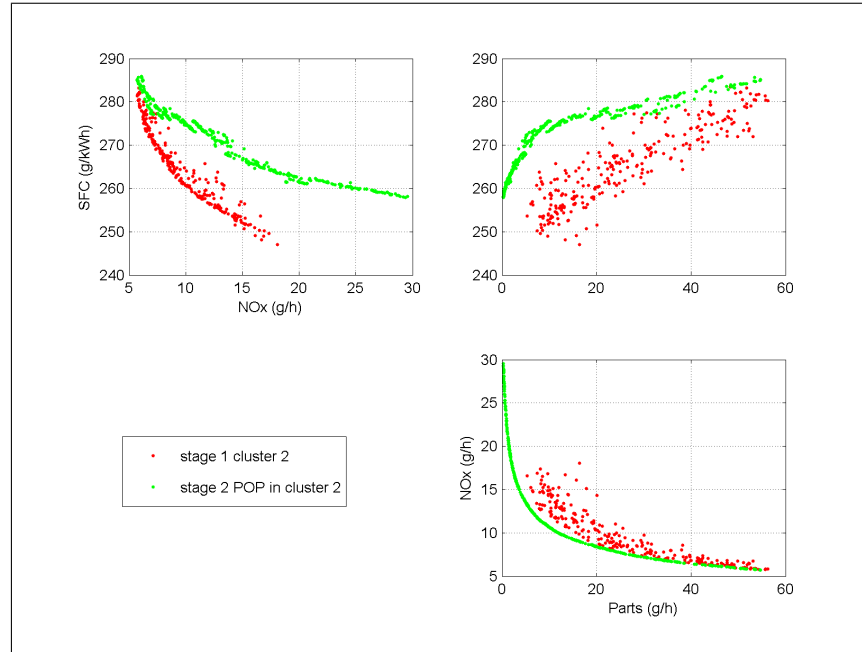


Figure 4.9: Scatter plot of Cluster 2 and the Stage 2 POP resulting from the (reduced) two-objective optimisation of NO<sub>x</sub> and Parts in Cluster 2.

### Selection of Final Solution and Conclusions from the Objective Reduction Process

Since with a three-objective problem it is straightforward to visualise, analyse and interpret pairwise scatter plots, these were used to select a final solution. With higher dimensional problems parallel coordinates plots are easier to use, in which case the axes limits can be reduced to focus on preferred solutions. Selection of a final solution depends on which cluster is selected and what, if any, objective priorities are applied, with examples given below. In each case, the Data Cursor feature in the Matlab<sup>®</sup> Figure environment was used to identify reasonable compromise solutions, which have been tabulated in Table 4.6. Figure 4.10 is an example of the pairwise scatter plot for the first case with the preferred solution identified using the Data Cursor.

As Table 4.6 shows, some improvement in the retained Objectives, NO<sub>x</sub> and Parts, at the expense of the discarded Objective, SFC, has been achieved in the Stage 2 POP in Cluster 2 compared to the solution selected in Stage 1 Cluster 1, where, in contrast, the SFC is lower at the expense of NO<sub>x</sub> and Parts. In conclusion, the use of clustering, PCA and objective preference articulation has allowed the dimensionality

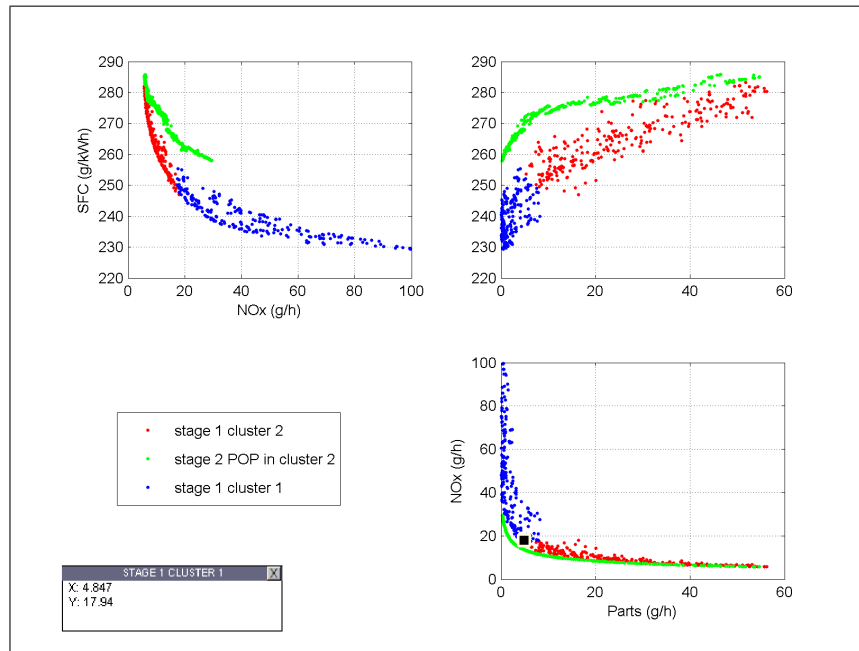


Figure 4.10: Scatter plot of Stage 1 Clusters 1 and 2 and the Stage 2 POP resulting from the (reduced) two-objective optimisation of  $\text{NO}_x$  and Parts in Cluster 2. The solution selected using the Matlab<sup>®</sup> Data Cursor feature from the Stage 1 Cluster 1 data is shown in black in the bottom right plot

Table 4.6: Final solutions selected.

Data Group	SFC (g/kWh)	$\text{NO}_x$ (g/h)	Parts (g/h)
Stage 1 Cluster 1	251.7	17.9	4.8
Stage 2 POP in Cluster 2	265.4	16.4	2.7

of the problem to be reduced to enable the search for preferred solutions to progress. Nevertheless, it was important to show in the further optimisation in Cluster 2, that while the ellipsoidal cluster boundary did allow the discarded Objective, SFC, to deteriorate slightly, the harmony with the Parts Objective was maintained.

## 4.8 Summary

A Many-Objective Optimisation Decision Making (MOODM) Process has been introduced. This comprises the elements of optimisation, partitioning the Pareto-Optimal Population (POP) into clusters, and objective reduction using principal component analysis (PCA) based rules. If objective reduction is achieved within a cluster, further optimisation with reduced objectives is carried out, subject to a hyper-ellipsoidal cluster boundary constraint in an attempt to preserve objective correlations.

The optimisation element utilised the widely used NSGAII algorithm (Deb, Pratap, Agarwal and Meyarivan, 2002), which was modified to incorporate the Progressive Preference Articulation method of Fonseca and Fleming (1998a) (PPA<sub>FF</sub>). This allows objective goals and priorities to be intuitively and efficiently specified and shrinks the search space to the DMs region of interest. The clustering element makes use of the k\*-Means algorithm (Cheung, 2003), which automatically determines the number of clusters in the POP. The clusters represent groups of like-solutions, within which any local objective harmony can be identified. This is achieved using principal component analysis (PCA) based rules designed to identify and retain only the most dominant and/or conflicting objectives.

The MOODM process is demonstrated using a simplified real-world example comprising a three-objective, single constraint Diesel Engine Base Calibration optimisation. The process resulted in no objective reduction in the first cluster and a reduction to a two-objective problem in the second cluster. A solution selected from the latter POP performed better in the retained objectives (and maintained correlation with the discarded objective) compared to the former POP.



## Chapter 5

# Case Study: Diesel Engine Base Calibration Optimisation

### 5.1 Introduction

This is the first of two automotive powertrain calibration optimisation case studies and concerns the optimisation of diesel base engine calibration (Cary, 2003). Vehicles equipped with diesel engine technology form approximately 50% of the European automotive market (Herzog *et al.*, 2007). This is due in no small part to the sharp rise in fuel prices and superior fuel economy offered by diesel technologies. As detailed in Section 2.9.3, diesel base engine calibration involves adjustment of the control actuator settings to achieve optimal trade-offs between competing objectives. Such objectives include fuel consumption as well as legislated emissions and combustion noise measures. Therefore, diesel base engine calibration represents a fundamentally important task for manufacturers in a very competitive market.

The six objective problem formulated in this case study was first introduced in Chapter 4 where it was simplified into a three objective problem to demonstrate the proposed objective reduction process. After specifying the optimisation problem and running the optimisation, a clustering analysis is carried out to partition the Pareto-optimal front. This is followed by a principal component analysis (PCA) and application of objective reduction rules to each cluster. These are used to exploit any local objective harmony and to identify the main conflicting objectives. Preference articulation is used not only to direct the search to a region of interest, but also to

discriminate between the main conflicting objectives, so that only the highest priority conflicting objectives are retained. Subsequent optimisation is carried out with reduced objectives to allow final preferred solutions to be selected and compared. Further studies are conducted to compare this approach to alternative methods. These comprise applying the PCA and objective reduction process on the whole population, varying the threshold for selecting Principal Components and finally, an exclusive application of the Progressive Preference Articulation (PPA) method of Fonseca and Fleming (1998a).

## 5.2 Six Objective Diesel Problem Formulation

The full six-objective diesel problem as described in section 4.7.1 requires the minimisation of the following objectives:

- SFC (g/kWh).
- NO<sub>x</sub> (g/h).
- Parts (g/h).
- HC (g/h).
- CO (g/h).
- Noise (dBA).

The model boundary constraint, the parameter settings for NSGAI, the decision variables and their bounds were the same as for the three-objective problem as defined in Section 4.7.1. The exceptions were that the population size was increased to 4000 and the number of generations was increased to 5000 to provide a reasonable computational effort. The computational expense was considered acceptable, *i.e.* the optimisation took approximately three hours on a PC with an Intel® Core™ 2 Duo CPU T8300 2.40 GHz processor.

All the objectives were minimised subject to the constraint. Realistic preferences were applied to take into account that the emissions Objectives (NO<sub>x</sub>, Parts, HC and CO) have legislated limits and are therefore considered as being more important than SFC or noise. Furthermore, of these emissions Objectives, NO<sub>x</sub> and Parts have greater

Table 5.1: Initial goals and priorities for the six-objective diesel problem, where the last row represents the constraint. The highest(lowest) Priority number indicates the highest(lowest) priority objective.

Objective	Goal	Priority
SFC	$-\infty$	1
NO <sub>x</sub>	$-\infty$	3
Parts	$-\infty$	3
HC	$-\infty$	2
CO	$-\infty$	2
Noise	3	1
model_bdry	0	4

priority than HC or CO. This is because the latter can be reduced using exhaust after-treatment systems, *e.g.* catalytic convertors, while the former have to be minimised at source (Brooks *et al.*, 2005). Nevertheless, it is still important to minimise HC and CO emissions as this will have a positive cost benefit to the design of the after-treatment system. Following the constrained minimisation problem formulation as defined by Fonseca and Fleming (1998a), this means that the constraint was assigned as the highest priority objective, with the emissions objectives at intermediate priorities and the SFC and noise objectives at the lowest priority. The priorities are listed in Table 5.1.

## 5.3 Six Objective Diesel Problem Results

### 5.3.1 Clustering and Verification

As with the three-objective problem, the Cluster verification Rules from Section 4.4.3 were applied to verify the number and location of clusters of like Pareto-Optimal solutions. The k\*-Means parameter settings were the same as used previously on the three-objective problem in Section 4.7.2.

In addition, the clustering has been repeated from various initial numbers of clusters and on different size Pareto-Optimal Population (POP) sizes with the results shown in Table 5.2 and described as follows:

- Applying CR1, robust clustering was repeated on the same population of 4000

Table 5.2: Comparison of clusters resulting from the POP of 4000 with those of randomly sub-sampled POPs.

Cluster Property	Population	Cluster 1						Cluster 2					
Centres	4000	251.2	31.8	14.8	6.5	59.4	1.2	282.2	8.4	71.0	6.9	392.0	0.5
	2000	251.2	31.7	14.8	6.5	59.4	1.2	282.5	8.3	72.0	6.9	400.5	0.5
	1000	251.2	31.5	14.7	6.5	58.8	1.2	281.2	8.7	69.9	6.9	385.8	0.5
	500	251.0	31.4	14.2	6.5	56.1	1.1	279.8	9.2	67.2	6.9	366.5	0.5
Correlation Matrices	4000	1						1					
		-0.98	1					-0.95	1				
		0.98	-0.93	1				0.90	-0.97	1			
		-0.70	0.78	-0.61	1			0.91	-0.79	0.68	1		
		0.94	-0.88	0.99	-0.52	1		0.88	-0.94	0.99	0.68	1	
		-0.93	0.97	-0.85	0.73	-0.78	1	-0.94	0.81	-0.76	-0.95	-0.77	1
	2000	1						1					
		-0.98	1					-0.95	1				
		0.98	-0.93	1				0.88	-0.96	1			
		-0.71	0.80	-0.62	1			0.91	-0.77	0.64	1		
		0.94	-0.87	0.99	-0.53	1		0.86	-0.93	0.99	0.63	1	
		-0.93	0.96	-0.85	0.76	-0.76	1	-0.93	0.80	-0.74	-0.95	-0.73	1
	1000	1						1					
		-0.98	1					-0.95	1				
		0.98	-0.94	1				0.90	-0.97	1			
		-0.74	0.83	-0.66	1			0.91	-0.78	0.67	1		
		0.95	-0.88	0.99	-0.57	1		0.88	-0.94	0.99	0.66	1	
		-0.94	0.97	-0.87	0.82	-0.79	1	-0.94	0.82	-0.76	-0.96	-0.76	1
	500	1						1					
		-0.99	1					-0.95	1				
		0.98	-0.95	1				0.91	-0.97	1			
		-0.77	0.85	-0.69	1			0.91	-0.79	0.71	1		
		0.95	-0.90	0.99	-0.62	1		0.89	-0.95	0.99	0.70	1	
		-0.95	0.98	-0.88	0.83	-0.82	1	-0.94	0.84	-0.78	-0.97	-0.78	1

Pareto-optimal solutions from an initial number of clusters,  $cs$ : 4, 5 three times, 6 and 8. All runs resulted in a final number of clusters,  $cf = 2$ .

- Applying CR2, one of the pairs of clusters (the second run from a  $cs = 5$ ) resulting from the application of CR1 to the POP of 4000, was randomly sub-sampled using the Matlab<sup>®</sup> routine *randperm* into smaller POPs of sizes: 4000, 2000, 1000 and 500. Robust clustering (retaining 95% of the data) was run on all of the POPs and all yielded  $cf = 2$ .
- Applying CR3,
  - The cluster centres for Objectives 1 to 6 were determined and in comparison were broadly similar for all POPs, as can be seen in Table 5.2.



However, closer examination of the centres, more so for Cluster 2, showed some divergence of the 500 POP centres from those of the 4000 POP.

- Since the PCA was based on the correlation matrices, these were calculated for all POPs and had broadly similar magnitude and structure, as can be seen in Table 5.2. Again, further inspection of the smaller POPs shows some fairly mild evidence of divergence from the correlation matrices of the 4000 POP.

Consequently, it was concluded that the POP of 1000 was the smallest POP providing adequate agreement with the POP of 4000.

Further evidence is provided in the parallel coordinates plots of the POP of 4000 and the sub-sampled POP of 1000, as displayed in Figures 5.1 and 5.2 respectively. Graphically, these plots show similar pattern, range and harmony/conflict relationships between objectives when the same objective order is considered. To consider all objective orders would require  $m(m-1)$  parallel coordinate plots, where  $m$  is the number of objectives, *i.e.* fifteen plots for six objectives. This would be very cumbersome in practice and is one of the weaknesses of such plots.

Table 5.3: Cluster data bounds for the selected POP of 1000.

		Cluster 1							Cluster 2						
Cluster Membership Count		673							327						
Cluster Bounds	Upper	264.3	65.2	36.2	6.8	145.5	3.0	297.9	16.2	103.6	7.7	807.8	1.0		
	Lower	235.0	14.6	1.2	6.1	25.2	0.7	261.1	6.3	28.3	6.3	93.5	0.0		

- Applying CR4 to the selected POP of 1000, Cluster 1 had 673 members while Cluster 2 had 327, so both had significant membership. In order to compare objective ranges in the POP of 1000, the upper and lower bounds of the data in each cluster were determined. In terms of minimising objectives with a lower and ideally smaller, range:

- Cluster 1 was better for Objectives 1 (SFC), 3 (Parts), 4 (HC) and 5 (CO).

- Cluster 2 was better for Objectives 2 (NO<sub>x</sub>) and 6 (noise).

Therefore both clusters were retained. These results are shown in Table 5.3.

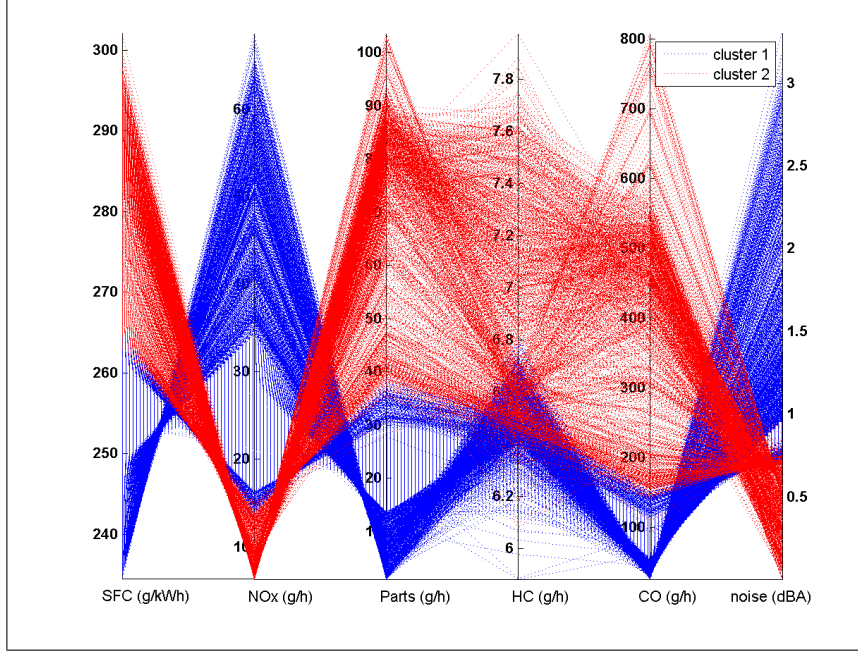


Figure 5.1: Parallel coordinates plot of the clusters in the original POP of 4000.

### 5.3.2 Principal Component Analysis and potential Objective Reduction

Just as with the three-objective problem, the process of Section 4.5 was applied to each cluster to identify any opportunity for objective reduction.

#### Objective Reduction Rules applied to Cluster 1

1. Applying OR1,  $\lambda_1/\lambda_6 = 21495$ , and this is taken as evidence that the last PC is a near-linear dependency. In other words, the eigenvector for the PC6 is  $v^6 = 0.0992Obj1 - 0.1851Obj2 - 0.8101Obj3 + 0.007Obj4 + 0.5464Obj5 - 0.0014Obj6$ , with the eigenvector coefficients for Objectives 4 and 6 being near zero, as shown in Table 5.4. Further examination revealed that of the eigenvector coefficients which are non near-zero, *i.e.* Objectives 1, 2, 3 and 5, Objective 1

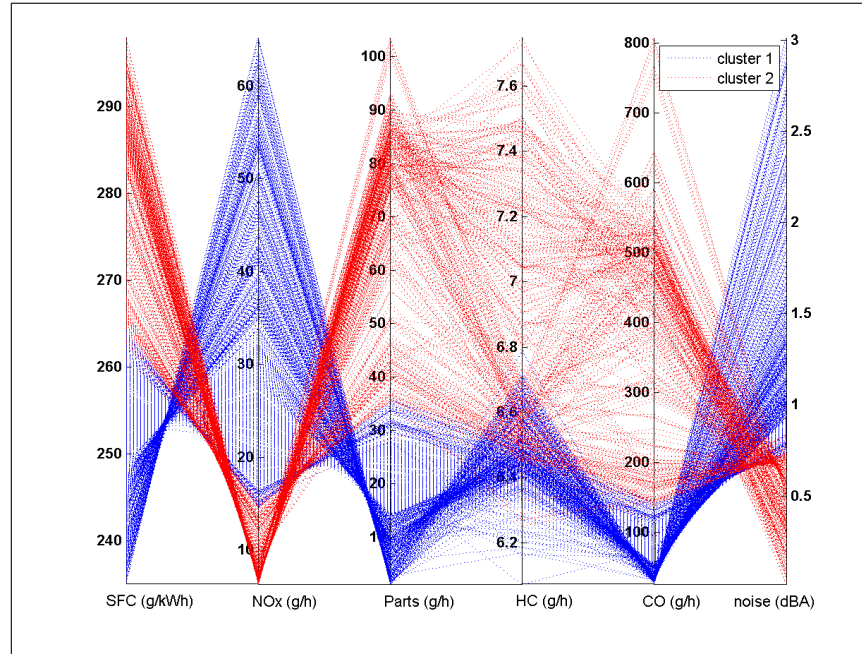


Figure 5.2: Parallel coordinates plot of the clusters in the sub-sampled POP of 1000.

Table 5.4: PCA results for Cluster 1 in the selected POP of 1000. The eigenvalues are listed in Principal Component order (largest first). As there is evidence of a near-linear dependency ( $\lambda_1/\lambda_6 \geq 10000$ ), the eigenvector of the last PC is examined for those objectives, which have non near-zero coefficients. Of these, Objective 1 (SFC) has the lowest priority (shown shaded) and was thus selected to be discarded.

		Cluster 1					
PC Eigenvalues ( $\lambda_p$ )		5.3229					
		0.5465					
		0.1189					
		0.0104					
		0.0010					
		0.0002					
Eigenvectors ( $\mathbf{v}^p$ )	Obj1	-0.4308	-0.1327	0.0804	-0.3422	0.8145	0.0992
	Obj2	0.4306	-0.0874	-0.1797	0.6727	0.5365	-0.1851
	Obj3	-0.4189	-0.3354	-0.1850	0.0594	-0.1343	-0.8101
	Obj4	0.3507	-0.7546	0.5348	-0.1384	-0.0493	0.0070
	Obj5	-0.3997	-0.4928	-0.3577	0.3779	-0.1643	0.5474
	Obj6	0.4132	-0.2236	-0.7163	-0.5146	0.0368	-0.0014

Table 5.5: Clustering and PCA results for the selected POP of 1000, where Cluster 1 has had the number of objectives reduced to five. For each cluster, the eigenvalues are listed in Principal Component order (largest first), followed by the corresponding cumulative percentage of total variation. Also, the eigenvectors and as appropriate, rotated eigenvectors, are listed by objective for the retained PCs with the selected significant eigenvector coefficients shown shaded.

		Cluster 1		Cluster 2	
PC Eigenvalues ( $\lambda_p$ )		n/a		5.3040	
		4.0112		0.6012	
		0.6458		0.0601	
		0.3322		0.0267	
		0.0104		0.0074	
		0.0004		0.0006	
Cumulative % of Total Variation		n/a		88.4	
		80.2		98.4	
		93.1		99.4	
		99.8		99.9	
		100		100	
		100		100	
PCs retained		PC1	PC2	PC1	PC2
Eigenvectors ( $\mathbf{v}^p$ )	Obj1	n/a	n/a	-0.4294	-0.1161
	Obj2	-0.4922	-0.0159	0.4217	-0.2271
	Obj3	0.4830	0.2355	-0.4083	0.4374
	Obj4	-0.3346	0.9124	-0.3816	-0.5969
	Obj5	0.4562	0.3251	-0.4036	0.4439
	Obj6	-0.4521	-0.0782	0.4032	0.4362
Rotated Eigenvectors ( $\mathbf{v}^{p'}$ )	Obj1	n/a	n/a	-0.2357	-0.3772
	Obj2	-0.4694	0.1491	0.4636	0.1205
	Obj3	0.5339	0.0611	-0.5968	0.0428
	Obj4	-0.0113	0.9717	0.1264	-0.6971
	Obj5	0.5385	0.1544	-0.5978	0.0507
	Obj6	-0.4523	0.0769	-0.0012	0.5940

(SFC) had the lowest priority and was therefore discarded. PCA was applied to the remaining five objectives, the results of which are shown in Table 5.5.

2. Applying OR1 to the remaining five objectives,  $\lambda_2/\lambda_6 = 8930.8$ , so there is not sufficient evidence of a near-linear dependency.
3. Applying OR2,  $t_2 \approx 95\%$ , *i.e.* retain the first two PCs, which account for approximately 95% of the cumulative percentage total variation. Note that two

PCs account for 93.1% cumulative variation, whereas three PCs account for 99.8%, which is much further away from 95%.

4. Applying OR3a), the eigenvectors for the two retained PCs were rotated and the threshold for the test of significance is  $5^{-0.5} = 0.4472$ .
5. Applying OR3c) to PC1, the rotated eigenvector coefficients,  $v_2^1$ ,  $v_3^1$ ,  $v_5^1$  and  $v_6^1$  exceed 0.4472 in magnitude and thus are significant.  $v_2^1$  and  $v_6^1$  have the same sign, but Objective 2 (NO<sub>x</sub>) has higher priority and so is retained. Likewise,  $v_3^1$  and  $v_5^1$  have the same sign, but Objective 3 (Parts) has higher priority and so is retained. Both of the retained objectives have their eigenvector coefficients shown shaded in Table 5.5.
6. Applying OR3b) to PC2, only the rotated eigenvector coefficient,  $v_4^2$  is significant and so the corresponding Objective 4 (HC) is also retained. On closer examination, it can be seen that the magnitude of  $v_3^2$  is nearly 1 and relatively small for the other objectives in PC2. Further, the corresponding eigenvector coefficient in PC1, *i.e.*  $v_4^1$  is close to zero in magnitude. This suggests that Objective 4 is independent, since it dominates PC2 and has negligible effect in PC1, can be removed from any further multi-objective optimisation in Cluster 1 and formulated as a single-objective optimisation. While one two-objective and a single-objective optimisation may be considered a simpler problem formulation, in the interests of consistency, it was decided to leave such a simplification for future work.

In summary, in Cluster 1 the objectives were reduced to just three: NO<sub>x</sub>, Parts and HC. Evidence of this can be seen in Figure 5.2, which shows harmony between SFC, Parts and CO and also between NO<sub>x</sub> and noise.

### Objective Reduction Rules applied to Cluster 2

1. Applying OR1 to the remaining five objectives,  $\lambda_1/\lambda_6 = 8412$ , so there is not sufficient evidence of a near-linear dependency.
2. Applying OR2,  $t_2 \approx 95\%$ , *i.e.* retain the first two PCs, which account for approximately 95% of the cumulative percentage total variation.

3. Applying OR3a), the eigenvectors for the two retained PCs were rotated and the threshold for the test of significance is  $6^{-0.5} = 0.4082$ .
4. Applying OR3c) to PC1, the rotated eigenvector coefficients,  $v_2^1$ ,  $v_3^1$  and  $v_5^1$  exceed 0.4082 in magnitude and thus are significant.  $v_3^1$  and  $v_5^1$  have the same sign, but Objective 3 (Parts) has higher priority and so is retained. Objective 2 is also retained as  $v_2^1$  is the only positive significant eigenvector coefficient. Both of the retained objectives have their eigenvector coefficients shown shaded in Table 5.5.
5. Applying OR3b) to PC2, only the rotated eigenvector coefficients,  $v_4^2$  and  $v_6^2$  are significant and so Objectives 4 (HC) and 6 (noise) are retained.

In summary, in Cluster 2 the objectives were reduced to four:  $\text{NO}_x$ , Parts, HC and noise. Unlike Cluster 1, evidence of this in Figure 5.2 is not clear. Solutions in Cluster 2 are well spread for each of the Objectives: Parts, HC and CO, so it is difficult to visually discern any correlations.  $\text{NO}_x$  appears to be correlated with noise, but the PCA and objective reduction rules have resulted in  $\text{NO}_x$  being selected in PC1 and noise being chosen in PC2. This is evidence that it may be difficult to identify objective correlations with parallel coordinates plots.

Table 5.6: Goals and priorities for the reduced, three-objective diesel problem in Cluster 1.

Objective	Goal	Priority
SFC	$-\infty$	0
$\text{NO}_x$	$-\infty$	2
Parts	$-\infty$	2
HC	$-\infty$	1
CO	$-\infty$	0
noise	3	0
model_bdry	0	3
cluster_bdry	0	3

### Further Optimisation within Cluster 1

In the previous section the objectives were reduced from six to three:  $\text{NO}_x$ , Parts and HC. The sub-sampled population resulting originally from the initial six-objective problem was used as the initial population for further optimisation. Also, as with the three-objective problem in Section 4.7.2, a hyper-ellipsoidal cluster boundary constraint, `cluster_bdry`, was used to keep the search within Cluster 1 and preserve objective correlations. The problem was formulated as before in Section 5.2, except that the number of generations was reduced to 2000, which was still a reasonable computational effort given the reduction in objectives. The goals and priorities are specified in Table 5.6, where the discarded Objectives (SFC, CO and noise) have a priority set to 0.

The resulting population is plotted against the original Cluster 1 in parallel coordinates format in Figure 5.3 and in scatter plot format in Figure 5.4. The parallel coordinates plot shows for the retained objectives, both  $\text{NO}_x$  and HC have improved, but partially at the expense of Parts (due to the fact that all three objectives

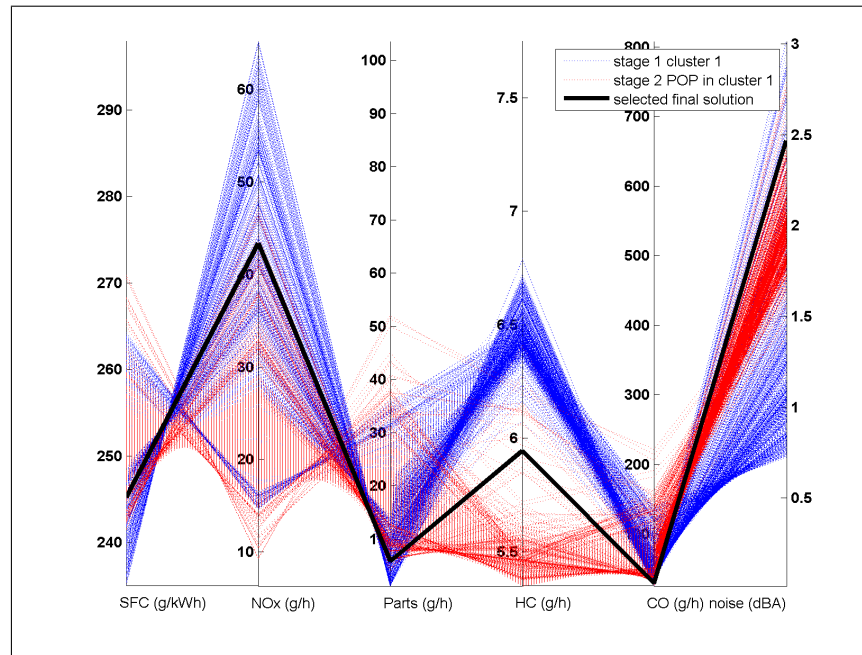


Figure 5.3: Parallel Coordinates plot of Cluster 1, the Stage 2 POP resulting from the (reduced) three-objective optimisation of  $\text{NO}_x$ , Parts and HC and the selected final solution.

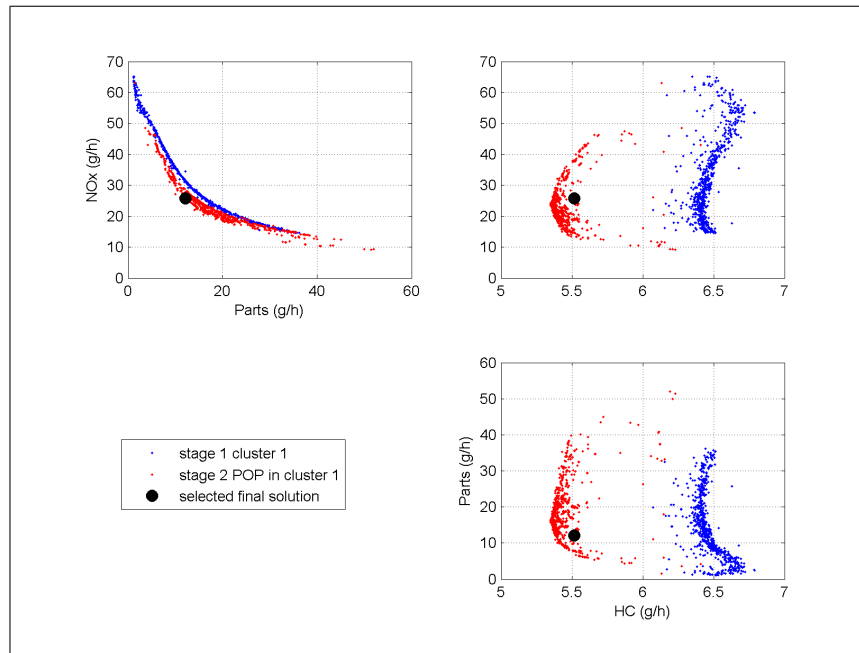


Figure 5.4: Scatter plot of Cluster 1, the Stage 2 POP resulting from the (reduced) three-objective optimisation of NO<sub>x</sub>, Parts and HC and the selected final solution.

conflict), while the discarded Objectives, SFC, CO and noise broadly show no deterioration. As with the three-objective problem, the plot indicates that the Stage 1 objective correlations have been maintained in Stage 2. The two-dimensional scatter plots show this more clearly and suggest that a trade-off solution for NO<sub>x</sub> and Parts can be chosen with simultaneously improved HC. The Data Cursor feature in the Matlab<sup>®</sup> Figure environment was used to select such a solution and hence determine from the final population the corresponding values for the discarded objectives, as shown in Table 5.8.

### Further Optimisation within Cluster 2

In Cluster 2 the objectives were reduced from six to four, *i.e.* NO<sub>x</sub>, Parts, HC and noise. The sub-sampled population resulting originally from the initial six-objective problem was used as the initial population for further optimisation and an ellipsoidal cluster bound constraint was used to keep the search within Cluster 2 and preserve objective correlations. The problem was formulated as before in Section 5.2 and run for 2000 generations. The goals and priorities are specified in Table 5.7, where the discarded Objectives (SFC and CO) have a priority set to 0.



Table 5.7: Goals and priorities for the reduced, four-objective diesel problem in Cluster 2.

Objective	Goal	Priority
SFC	$-\infty$	0
NO <sub>x</sub>	$-\infty$	3
Parts	$-\infty$	3
HC	$-\infty$	2
CO	$-\infty$	0
noise	3	1
model_bdry	0	4
cluster_bdry	0	4

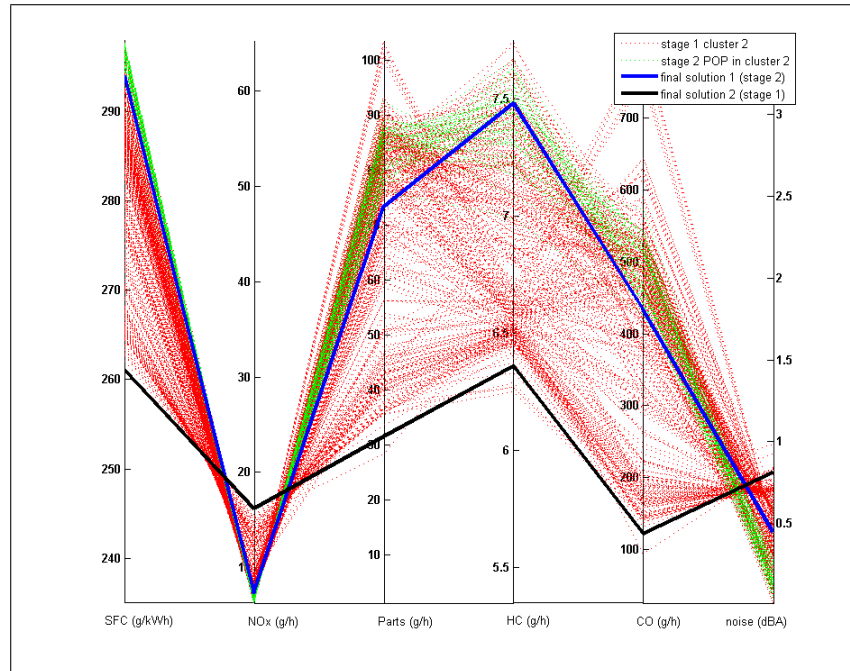


Figure 5.5: Parallel Coordinates plot of Cluster 2, the Stage 2 POP resulting from the (reduced) four-objective optimisation of NO<sub>x</sub>, Parts, HC and noise and selected final solutions from stage 1 and stage 2.

The resulting population is plotted against the original Cluster 2 in parallel coordinates format in Figure 5.5 and in scatter plot format in Figure 5.6. The parallel coordinates plot shows for the retained objectives, that  $\text{NO}_x$  has improved along with noise at the expense of Parts and HC, whilst the discarded Objectives (SFC and CO) have both maintained their objective correlations. In summary, no overall improvement was achieved with this objective reduction.

This is even more clear in the two-dimensional scatter plots. Again, the Data Cursor feature was used to select solutions. Firstly, a trade-off solution ('final solution 1') in terms of the higher priority Objectives,  $\text{NO}_x$  and Parts, was chosen based on lowest Parts. As can be seen from the figure, an improved solution ('final solution 2') in terms of Parts can be found in the Cluster 2 from stage 1, albeit that this is at the expense of  $\text{NO}_x$ . Both of these solutions are displayed in Table 5.8

### 5.3.3 Conclusions from the Objective Reduction Process

Table 5.8 summarises the results from the objective reduction process applied in each of the two clusters. Of the two highest priority Objectives ( $\text{NO}_x$  and Parts), final

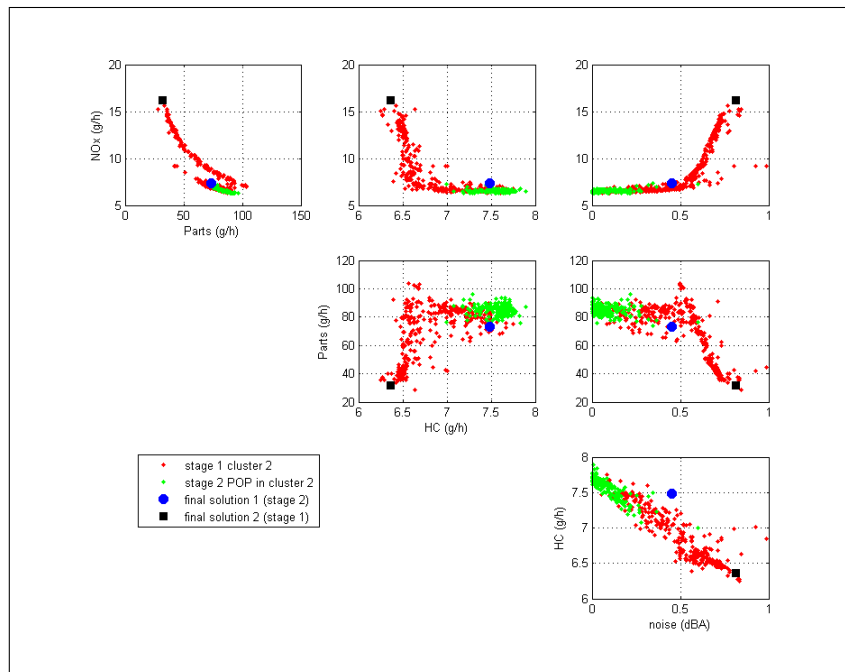


Figure 5.6: Scatter plot of Cluster 2, the Stage 2 POP resulting from the (reduced) four-objective optimisation of  $\text{NO}_x$ , Parts, HC and noise and selected final solutions from stage 1 and stage 2.

Table 5.8: Final solutions selected.

Stage	Cluster	Final Solution	SFC (g/kWh)	NO <sub>x</sub> (g/h)	Parts (g/h)	HC (g/h)	CO (g/h)	noise (dBA)
2	1	1	249.1	25.8	12.1	5.5	44.7	2.1
2	2	1	294.0	7.3	73.3	7.5	434.0	0.5
1	2	2	261.1	16.2	31.6	6.4	122.2	0.8

solution 1 in Cluster 2 has achieved the lowest NO<sub>x</sub>, but this was at the expense of Parts and deteriorated SFC, HC and especially CO. However, in practice, diesel calibrators tend to prioritise Parts higher (*i.e.* is more important) than NO<sub>x</sub>. In this case, final solution 2 in Cluster 2 is improved for SFC, Parts, HC and CO and only marginally deteriorated for noise. By comparison, final solution 1 in Cluster 1 is further improved in Parts (again at the expense of NO<sub>x</sub>), SFC, HC and CO and although noise has deteriorated, it is still within the 3 dBA constraint.

An iterative, complexity reduction process comprising the following stages was used in this case study:

- **Multi-Objective Optimisation** - using the NSGAII modified to incorporate the Progressive Preference Articulation (PPA) approach of Fonseca and Fleming (1998a) to specify objective priorities and goals to direct the search.
- **Clustering** - to partition the Pareto-Optimal Population to allow any local objective harmony to be exploited.
- **PCA and Objective Reduction** - to identify opportunities for objective reduction within each cluster with the possible application of PPA to discriminate between the main conflicting objectives.

In summary, the original six-objective optimisation has been split into two smaller optimisation problems via the complexity reduction process summarised above. Of these two sub-problems and with further preference articulation, the three-objective sub-problem was able to provide an improved solution in comparison to the four-objective sub-problem.

## 5.4 Objective Reduction applied without Clustering

The application of clustering as used in the previous section poses an additional stage in the decision-making process. In addition, PCA and objective reduction rules have previously been applied to the final population, *i.e.* without clustering (Deb and Saxena, 2005). Therefore, it was of interest to explore the latter approach and compare results on the six-objective diesel problem.

The PCA and objective reduction process of Section 4.5 was applied to the final population resulting from the optimisation as specified in Section 5.2 to identify any opportunity for objective reduction. The results are shown in Table 5.9. No attempt was made to follow the PCA and objective reduction process defined in Deb and Saxena (2005), since the authors stated that this approach showed some vulnerability in that not all conflicting objectives could be correctly identified.

1. Applying OR1,  $\lambda_1/\lambda_6 = 2302$ , so there is not sufficient evidence of a near-linear dependency.
2. Applying OR2,  $t_2 \approx 95\%$ , *i.e.* retain the first two PCs, which account for approximately 95% of the cumulative percentage total variation.
3. Applying OR3a), the eigenvectors for the two retained PCs were rotated and the threshold for the test of significance is  $6^{-0.5} = 0.4082$ .
4. Applying OR3b) to PC1, the rotated eigenvector coefficients,  $v_3^1$ ,  $v_4^1$ , and  $v_5^1$  exceed 0.4082 in magnitude and thus are significant. Of this group,  $v_4^1$  has the largest magnitude, is significantly larger than  $v_3^1$  or  $v_5^1$  (even though Objective 3 has higher priority) and so the corresponding Objective 4 (HC) is retained (shown shaded in Table 5.9).
5. Applying OR3b) to PC2, the rotated eigenvector coefficients,  $v_2^2$ ,  $v_4^2$  and  $v_6^2$  exceed 0.4082 in magnitude and thus are significant. Of this group,  $v_2^2$  is largest in magnitude and priority and so the corresponding Objective 2 (NO<sub>x</sub>) is retained.

In summary, the application of PCA and the objective reduction rules has resulted in the number of objectives being reduced from six in the original final population to

Table 5.9: PCA results for the final POP without clustering. The eigenvalues are listed in Principal Component order (largest first), followed by the corresponding cumulative percentage of total variation. Also, the eigenvectors and as appropriate, rotated eigenvectors, are listed by objective for the retained PCs with the selected significant eigenvector coefficients shown shaded.

PC Eigenvalues ( $\lambda_p$ )		4.9241	
		0.7791	
		0.2546	
		0.0354	
		0.0048	
		0.0021	
Cumulative % of Total Variation		82.1	
		95.1	
		99.3	
		99.9	
		100	
		100	
PCs retained		PC1	PC2
Eigenvectors ( $\mathbf{v}^p$ )	Obj1	-0.4485	-0.0121
	Obj2	0.4060	-0.4761
	Obj3	-0.4396	-0.0814
	Obj4	-0.3226	-0.7226
	Obj5	-0.4217	-0.2242
	Obj6	0.3985	-0.4406
Rotated Eigenvectors ( $\mathbf{v}^{p'}$ )	Obj1	-0.3831	0.2335
	Obj2	0.0822	-0.6203
	Obj3	-0.4133	0.1705
	Obj4	-0.6635	-0.4314
	Obj5	-0.4758	0.0409
	Obj6	0.0951	-0.5864

two. This is considered by the author to be a somewhat drastic objective reduction in a single step.

Nevertheless, a further optimisation was conducted with the two remaining Objectives,  $\text{NO}_x$  and HC. The goals and priorities are specified in Table 5.10, where the discarded Objectives (SFC, Parts, CO and noise) have a priority set to 0. The initial population was the final population resulting from the original six-objective

Table 5.10: Goals and priorities for the reduced, two-objective diesel problem resulting from the application of PCA and objective reduction rules.

Objective	Goal	Priority
SFC	$-\infty$	0
NO <sub>x</sub>	$-\infty$	2
Parts	$-\infty$	0
HC	$-\infty$	1
CO	$-\infty$	0
noise	3	0
model_bdry	0	3

optimisation. The parameter settings were as before (Section 5.2), but run for 2000 generations. Unlike with the clustering approach, no constraint was applied in an attempt to preserve objective correlations.

The resulting population is plotted against the original population in parallel coordinates format in Figure 5.7. The plot shows that while the retained Objective

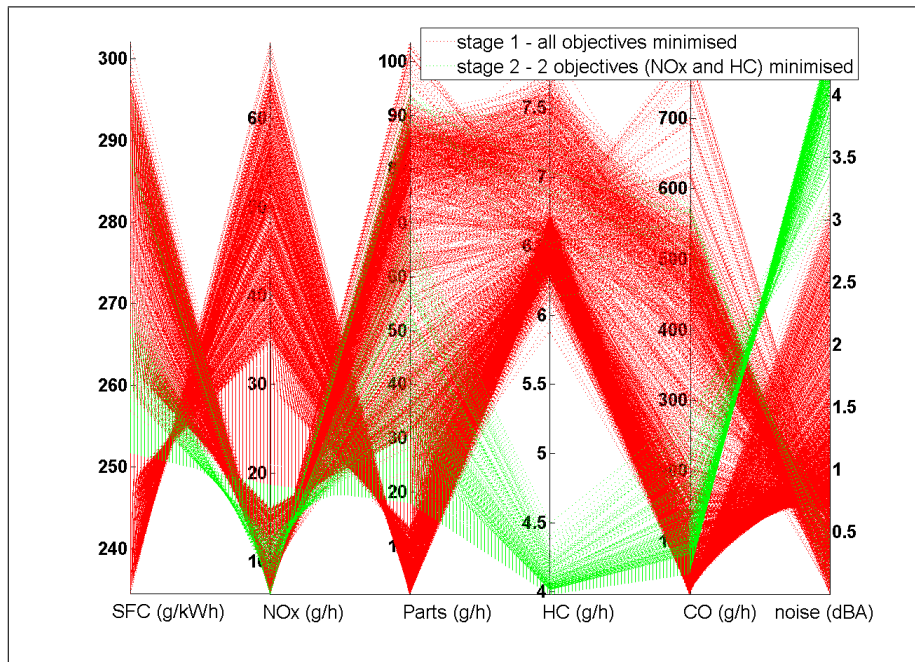


Figure 5.7: Parallel Coordinates plot of the original POP where all 6 objectives were minimised and the POP resulting from the minimisation of 2 Objectives (NO<sub>x</sub> and HC), which were retained based on PCA and objective reduction only, *i.e.* no clustering.

HC has improved, the other retained Objective,  $\text{NO}_x$ , has not. Furthermore, this improvement in HC has come at the expense of a deterioration in a high priority Objective, namely Parts, and also noise, where it can be seen that most of the population exceeds 3 dBA, which was the noise goal in the original six-objective formulation. In other words, for the most part the noise constraint has been violated. This can more clearly be seen in Figure 5.8, where there is a clear contrast with respect to the noise objective of the retained two-objective POP versus the best cluster identified when clustering, PCA and objective reduction were applied. In the latter case, an important difference is that the Cluster 1 hyper-ellipsoidal constraint was applied in an attempt to preserve correlations. Those solutions from the retained two-objective POP, which did satisfy the 3 dBA noise constraint compared unfavourably to those from the best cluster identified when clustering, PCA and objective reduction were applied with regard to all (discarded) objectives with the exception of  $\text{NO}_x$ . In conclusion, the use of clustering has allowed objective harmony

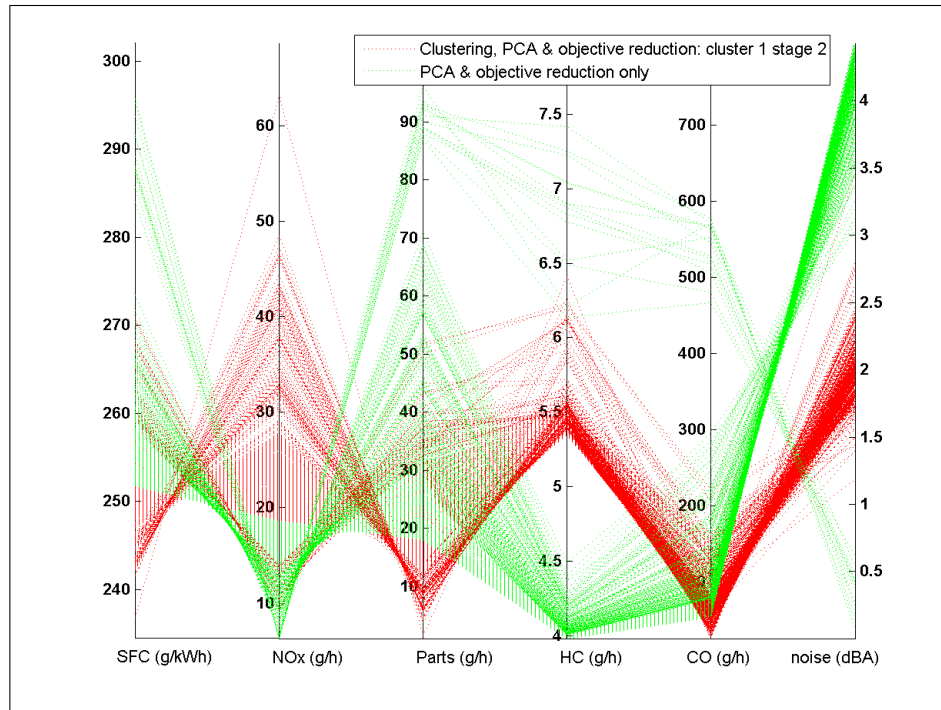


Figure 5.8: Parallel Coordinates plot of the Cluster 1 stage 2 POP resulting from clustering, PCA and objective reduction to give three retained Objectives ( $\text{NO}_x$ , Parts and HC) overlaid on the POP resulting from the minimisation of 2 Objectives ( $\text{NO}_x$  and HC), which were retained based on PCA and objective reduction only, *i.e.* no clustering.

to be exploited to reduce objectives and search for improved solutions whilst applying a cluster-based constraint to try and preserve objective correlations. This process has resulted in a reasonable compromise between the high priority objectives subject to the noise constraint as opposed to the process without clustering, which has reduced the objectives, but ignored a high priority objective, mostly violated the noise constraint and compared unfavourably with regard to the other objectives.

## 5.5 The Cut-off Threshold for selecting Principal Components

As stated in Section 3.5.2, the choice of the cut-off threshold in terms of cumulative percentage of variance is problem dependent and typically in the range of 70-90% (Jolliffe, 2002). One question which arises is that of which value to choose and what effect this has on objective reduction. The six-objective diesel problem can be used to answer this using a couple of alternative cut-off thresholds applied to the PCA results in Table 5.5.

- **Cut-off Threshold = 70%**

This would result in the first PCs only, being chosen in Clusters 1 and 2 using OR2 and thus the eigenvectors could not be rotated. Applying OR3c) to Cluster 1 results in eigenvector coefficients  $v_2^1$  and  $v_3^1$  being selected as these are significant and have the highest magnitude and priority. Applying OR3c) to Cluster 2 also gives  $v_2^1$  and  $v_3^1$  as both are significant, the former is the largest negative eigenvector coefficient and the latter has higher priority than  $v_1^1$ . In summary, in both clusters the problem has been reduced in dimension from six objectives to two, which is a somewhat drastic objective reduction. Furthermore, if only one PC is selected based on this threshold, as is the case with the three- and six-objective diesel problems, rules OR3b) and OR3c) dictate that the problem will be reduced to either one or two objectives, which is a severe dimension reduction.

- **Cut-off Threshold = 100%**

This means all the PCs would be chosen. When the objective reduction rules are applied then it is possible that no objective reduction will be achieved. This



Table 5.11: PCA results on the two clusters resulting from the selected POP of 1000, where Cluster 1 has had the number of objectives reduced to five. For each cluster, the resulting eigenvectors are displayed where the shaded coefficients are those selected from application of the objective reduction rules.

		Cluster 1					Cluster 2					
PCs retained		PC2	PC3	PC4	PC5	PC6	PC1	PC2	PC3	PC4	PC5	PC6
Eigenvectors ( $\mathbf{v}^p$ )	Obj1	n/a	n/a	n/a	n/a	n/a	-0.4294	-0.1161	0.2817	0.4560	-0.7014	-0.1513
	Obj2	-0.4922	-0.0159	-0.2553	-0.7829	-0.2816	0.4217	-0.2271	-0.6236	-0.0552	-0.5613	0.2522
	Obj3	0.4830	0.2355	-0.2912	0.0701	-0.7884	-0.4083	0.4374	-0.0891	0.0038	-0.0275	0.7958
	Obj4	-0.3346	0.9124	0.1999	0.1249	0.0049	-0.3816	-0.5969	0.1473	-0.6695	-0.0759	0.1493
	Obj5	0.4562	0.3251	-0.5365	-0.3152	0.5468	-0.4036	0.4439	-0.4756	-0.3670	-0.1464	-0.5077
	Obj6	-0.4521	-0.0782	-0.7226	0.5168	0.0126	0.4032	0.4362	0.5253	-0.4539	-0.4062	0.0141

is demonstrated with the six-objective problem with the results shown in Table 5.11, where application of the objective reduction rules has resulted in all six objectives being retained as shown by the shaded coefficients.

As shown with the six-objective problem, neither extreme of a 70 or 100% threshold is suitable for acceptable progressive dimension reduction. What is sought is a threshold where the majority of the variation is retained, but the objective reduction is not too drastic. In this case, a threshold of 95% seems a reasonable compromise and has given acceptable dimension reduction for both the three and six-objective problems.

As an insight into the objective dimension reduction opportunity, the impact of cut-off threshold on the number of selected PCs is shown in Figure 5.9. It can be seen

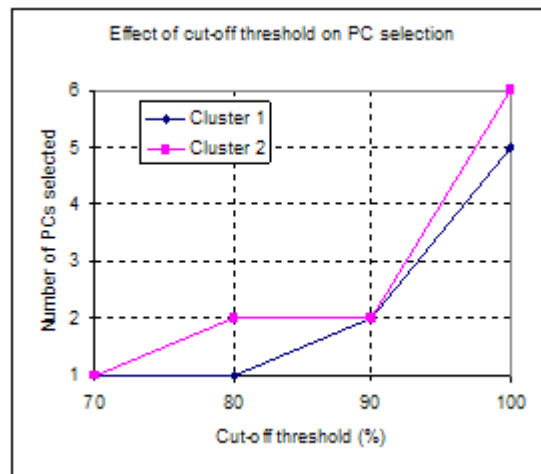


Figure 5.9: Effect of cut-off threshold on the number of PCs selected.

that a cut-off of approximately 95% gives a progressive reduction in the number of PCs and via the objective reduction rules, a potentially similar reduction in the number of objectives (see discussion under 'Proportion of Variance' in Section 3.5.2 together with guidance on the choice of threshold, which is provided later).

## 5.6 An Application of Progressive Preference Articulation

One alternative method used as a countermeasure for the lack of effective search in many-objective optimisation (Fleming *et al.*, 2005) is the Progressive Preference Articulation technique of Fonseca and Fleming (1998a) (PPA<sub>FF</sub>).

This approach has been initiated with the sub-sampled POP of size 1000 as described in Section 5.3.1. This sub-sampled POP was derived from the original POP of size 4000, which resulted from the optimisation formulated in Section 5.2 and was chosen to aid visualisation. A Parallel Coordinates plot of this sub-sampled POP is shown in Figure 5.10, from which there is some suggestion of harmony (objective pairs improve/deteriorate together) between SFC and Parts and between NO<sub>x</sub> and noise. This is much more evident in Figure 5.11 where the same data is shown, but with the

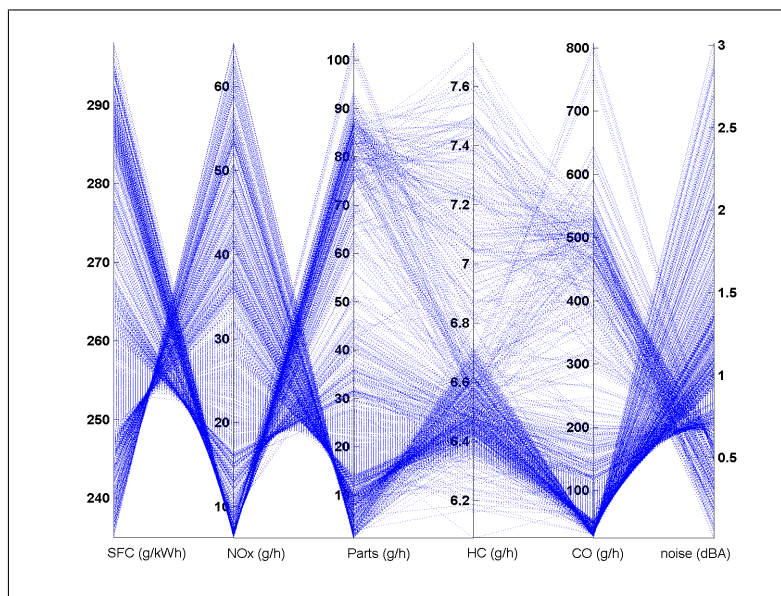


Figure 5.10: Parallel Coordinates plot of the sub-sampled POP of 1000 from the original POP of 4000.

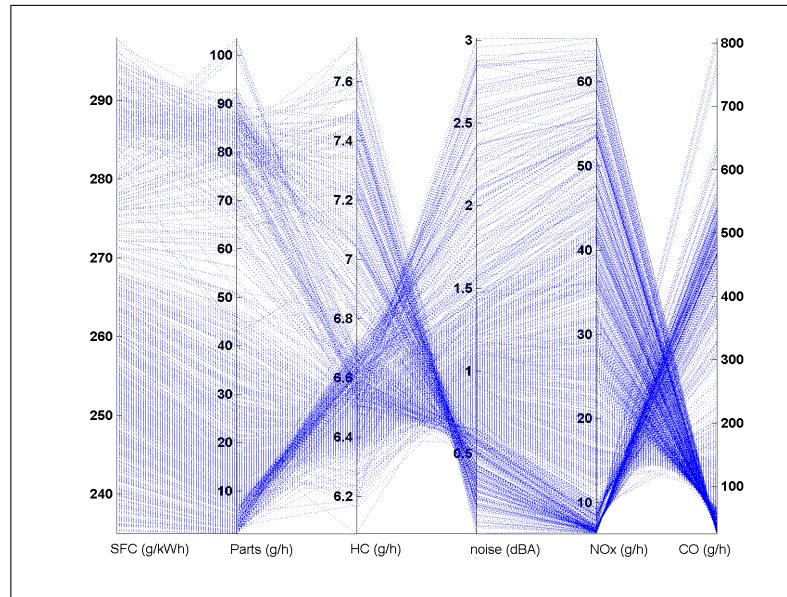


Figure 5.11: Same data as Figure 5.10, but with with re-ordered objectives to show evidence of harmony between SFC and Parts and between noise and  $\text{NO}_x$ .

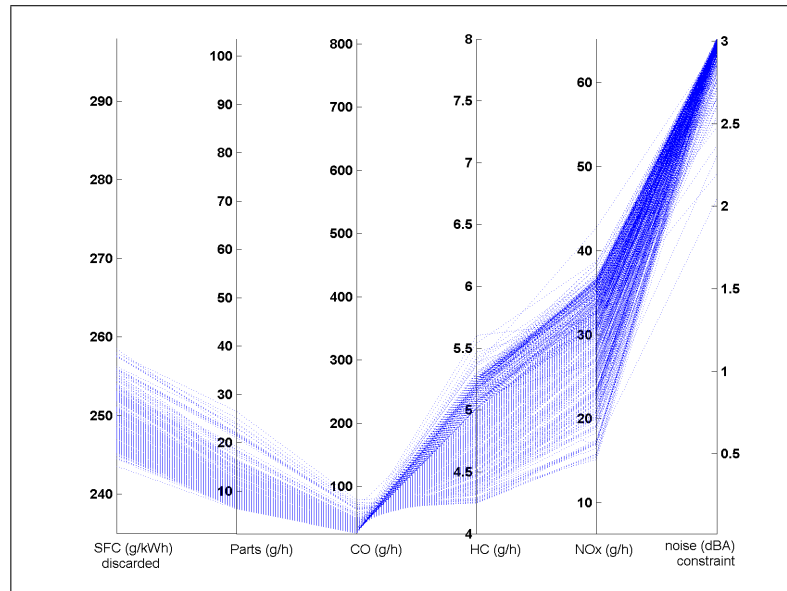


Figure 5.12: The population resulting from further optimisation with SFC discarded and noise changed to a constraint showing evidence of harmony between Parts and CO. Note: the objectives have been re-ordered.

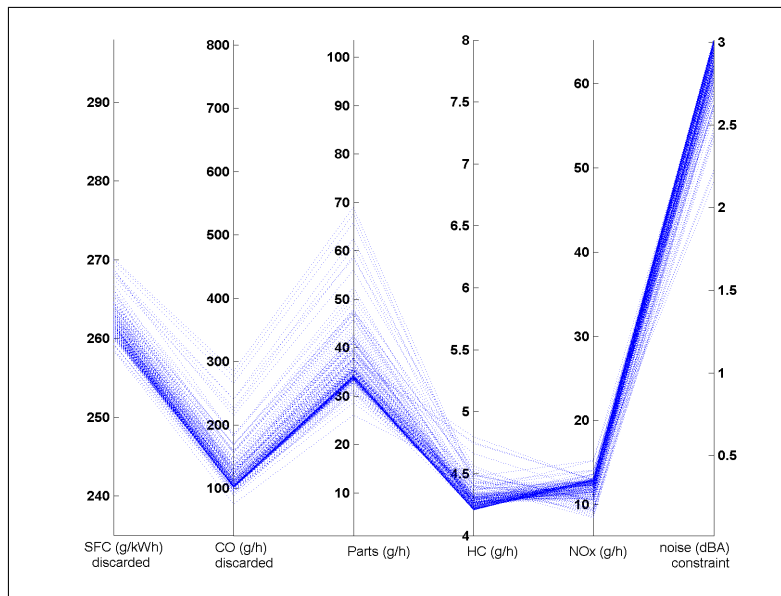


Figure 5.13: The population resulting from further optimisation with SFC and CO discarded and noise changed to a constraint. Note: the objectives have been re-ordered.

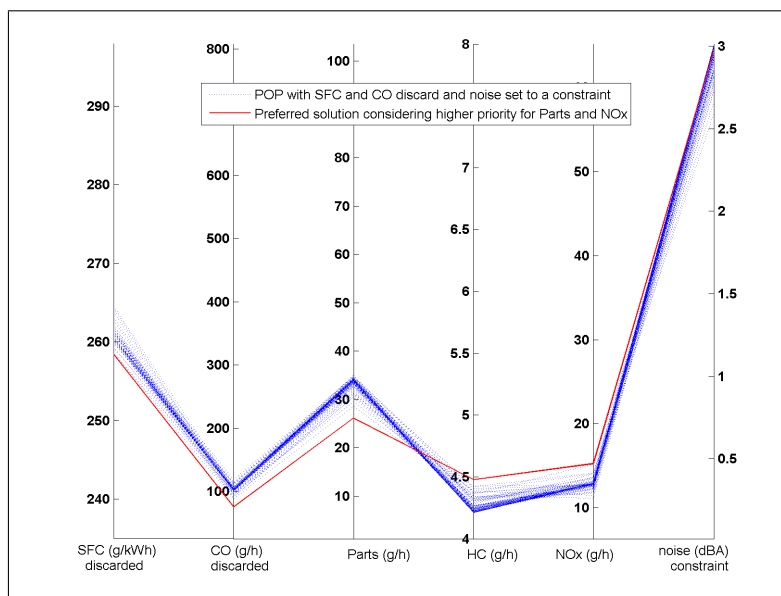


Figure 5.14: The same data as Figure 5.13, but with the population filtered for reduced Parts (blue dashed lines) and a preferred lowest Parts solution selected (red solid line).

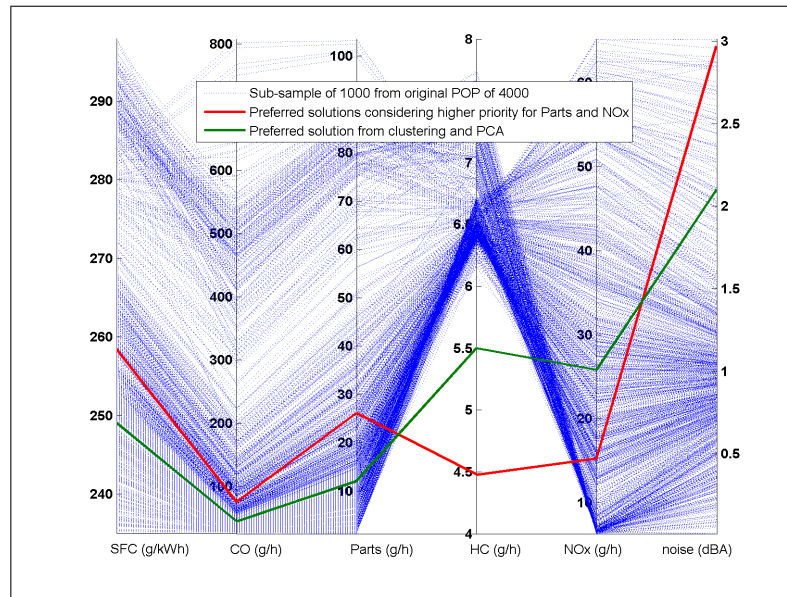


Figure 5.15: The same starting POP of 1000 as in Figure 5.10, but with re-ordered objectives (blue), the preferred solution from Figure 5.14 (red) and the preferred solution from clustering/PCA (green).

objectives re-ordered so that the harmonious ones are adjacent and the lines between them are somewhat parallel. The subsequent steps to identify an optimal solution are listed below and refer to Table 5.12, which details the decision maker's preferences in terms of goals and priorities.

### 1. 1st Objective Reduction

Table 5.12: Goals and priorities for original and subsequent reduced objective optimisations using Progressive Preference Articulation.

Objectives	Original Optimisation		1st Objective Reduction		2nd Objective Reduction	
	Goal	Priority	Goal	Priority	Goal	Priority
SFC	$-\infty$	1	$-\infty$	0	$-\infty$	0
NO <sub>x</sub>	$-\infty$	3	$-\infty$	2	$-\infty$	2
Parts	$-\infty$	3	$-\infty$	2	$-\infty$	2
HC	$-\infty$	2	$-\infty$	1	$-\infty$	1
CO	$-\infty$	2	$-\infty$	1	$-\infty$	0
noise	3	1	3	3	3	3
model_bdry	0	4	0	3	0	3

With reference to Figure 5.11 and the harmonious objectives, SFC can be discarded as it has a lower priority than Parts and although noise has a lower priority than  $\text{NO}_x$ , the noise objective has a goal of 3, which must be achieved and so was converted to a constraint. According to the PPA method of Fonseca and Fleming (1998a) ( $\text{PPA}_{\text{FF}}$ ), this is realised by increasing the priority of the noise objective (in this case from 1 to 3), so that solutions must satisfy this constraint first. A subsequent optimisation was carried out, formulated as before (Section 5.3.2), but run for 2000 generations with the goals and priorities as specified in Table 5.12.

## 2. 2nd Objective Reduction

The results of the 1st Objective Reduction are plotted in Figure 5.12 and show evidence of harmony between Parts and CO, with a suggestion that HC and  $\text{NO}_x$  are also harmonious. The discarded Objective SFC has been included in the plot to demonstrate that after further optimisation, it is still in harmony with Parts and likewise with the noise constraint and  $\text{NO}_x$ . It was decided to discard CO as it is lower priority than Parts and consistent with a progressive approach, retain HC, which is a lower priority than  $\text{NO}_x$ . A subsequent optimisation was carried out, formulated as the 1st Objective Reduction with goals and priorities as specified in Table 5.12. The results are plotted in Figure 5.13, which show that the harmony between the discarded Objectives of SFC and CO with Parts has been preserved and that there is some evidence of conflict between Parts, HC and  $\text{NO}_x$ . It can also be observed that the effect of discarding CO has significantly decreased  $\text{NO}_x$  at the expense of Parts, which is further evidence of the conflict between these objectives.

## 3. Selection and Comparison of a Final solution

Of the two highest priority Objectives, Parts and  $\text{NO}_x$ , as previously stated in Section 5.3.3, diesel calibrators attach more importance to Parts and so it was decided to filter the solutions to identify those with a lower Parts value (less than 35 g/kWh). This resulted in the plot in Figure 5.14, from which a single preferred solution in terms of lowest Parts was selected, shown in red. It is of interest to compare this solution in the context of the sub-sampled POP of 1000 and also in comparison with the selected solution (shown in green) resulting

from the clustering and PCA approach as plotted in Figure 5.15.

Several conclusions can be drawn from this comparison:

- Both the PPA<sub>FF</sub> and clustering/PCA approaches have explored a new region of objective space as demonstrated by the distinctly lower HC value of the selected solutions compared to the sub-sampled POP of 1000.
- The clustering/PCA solution is broadly comparable with that achieved by PPA<sub>FF</sub> and provides evidence of the efficacy of the former method.
- PPA<sub>FF</sub> does not exploit any local harmony for potential objective reduction.
- PPA<sub>FF</sub> relies on the DM being able to visually discern objective harmony and conflict from a parallel coordinates plot, which may be difficult and may involve some subjectivity. In contrast, the proposed clustering/PCA approach does not rely on visualisation, but instead utilises a set of rules and if specified, priorities to allow the DM to identify harmony and possible objective reduction.

## 5.7 Summary

A real-world diesel engine base calibration optimisation was carried out in this case study. This was a six-objective version of the problem introduced in Chapter 4. Likewise, the same principle of exploiting local harmony for objective reduction was used. A process of clustering and sub-sampling provided evidence of two clusters and that a Pareto-Optimal Population (POP) of 1000 was adequate, respectively. Objective reduction using PCA-based rules and objective priorities was performed in each of the two clusters, resulting in three- and four-objective sub-problems. Subsequent optimisation subject to a hyper-ellipsoidal cluster constraint was conducted to preserve objective correlations. In the three-objective sub-problem, a significant improvement was achieved in one of the retained objectives at very little cost to the other objectives. Whereas, in the four-objective problem, no further improvement was observed. With further preference articulation, preferred solutions were selected and that from the three-objective problem compared favourably to the solution generated from the four-objective problem.

Further studies were carried out to explore related aspects. Firstly, the objective reduction rules were applied without clustering, *i.e.* globally to the whole population. This resulted in the objective dimensionality being reduced to two. Subsequent optimisation resulted in a significant improvement in one of the retained objectives at the expense of a deterioration in a high priority, discarded objective. Furthermore, it could be seen that most of the resulting population violated a goal on another discarded objective and which was present in the original problem formulation. Unlike the clustering approach, it should be noted that there was no constraint to preserve correlations in subsequent optimisation with reduced objectives.

Secondly, the effect of varying the threshold for selecting PCs was explored. Two scenarios were considered. With a threshold of 70% only one PC and one objective was retained, which was considered too drastic an objective reduction. If a threshold of 100% was used, then all PCs and objectives were retained, *i.e.* there was no objective reduction. A threshold of 95% was considered a reasonable compromise.

Finally, the Progressive Preference Articulation method of Fonseca and Fleming (1998a) (PPA<sub>FF</sub>) was applied exclusively. A progressive approach of visually identifying objective harmony using parallel coordinate plots was used to reduce objectives. This process resulted in two objectives being discarded and one being converted to a constraint. Using the same objective priorities applied previously, a preferred solution was selected. This solution was broadly comparable to that generated from the clustering/PCA approach and showed the efficacy of the latter method. However, PPA<sub>FF</sub> relies on being able to visually discern objective harmony from a parallel coordinates plot, which may be difficult and involve some subjectivity. In many-objective problems, this may become an even greater challenge if large populations are required and need to be visualised. By comparison, the clustering/PCA approach did not rely on visualisation and instead used a set of rules and if specified, objective priorities, to allow the DM to identify local harmony and potential objective reduction. Exploiting potential local objective harmony in partitions of the Pareto-Optimal Population (POP) may provide more opportunities for complexity reduction than approaches applied to the whole POP.

A number of observations from this case study have relevance to higher dimensional optimisation problems:



- While this six-objective problem involved only one stage of objective reduction, it is possible that for problems with a larger number of objectives, the number of stages increases also. In such a scenario, the application of the clustering verification and objective reduction rules will become lengthy. A more compact form for these rules, which lends itself to being automated, would be useful.
- Higher dimensional problems may require larger populations to provide effective search. Larger populations in more objectives may generate more clusters. Both place significant demands on computational efficiency. Parallel computing is one approach to address this requirement.
- As the number of objectives increase so does the number of PCs. A principal component analysis on a larger number of objectives may reveal a finer graduation in the percentage of variation represented by the PCs. In other words, it may be possible that the threshold for selecting PCs could be varied slightly from the suggested 95% to retain a different number of PCs and potentially, a different degree of objective reduction.



## Chapter 6

# Case Study: Gasoline Engine Cold Start Calibration Optimisation

### 6.1 Introduction

With ever more stringent emission standards being imposed upon passenger vehicles, calibration approaches that minimise emissions during start and immediately after start or ‘run-up’ are becoming increasingly important to vehicle manufacturers (Wiemer *et al.*, 2007). Minimisation of hydrocarbon (HC) emissions during start-up, prior to the exhaust catalyst achieving a sufficiently high working temperature, is of particular significance (Bielaczyc and Merkisz, 1997; Shayler *et al.*, 1996).

During cold engine start, it is normal to inject excess fuel to account for unfavourable conditions that exist for air-fuel mixture preparation. With Gasoline Direct Injection (GDI) engine technology, as fuel is injected directly into the combustion chamber, the need to account for fuel films in the intake port (as in the case of a Port Fuel Injected (PFI) gasoline engine) is eliminated (Zhao *et al.*, 1999). This suggests that GDI technology should utilise less fuel during start-up than a PFI engine, and thereby offer greater potential for reducing HC emissions. Although some authors have reported savings in HC emissions of up to 50% for GDI over equivalent PFI engine technologies (Lee *et al.*, 2000), reductions of fuel quantity must not be achieved at the expense of decreased start and run-up quality.

Due to the many degrees of freedom associated with typical GDI controls, much more time and effort is needed to develop an engine start and run-up characterised by: an instantaneous first fire, followed by a stable and reliable engine run-up, while simultaneously minimising the delivered fuel quantity in an attempt to lower HC emissions. Apart from geometric design variables, such as the spark plug and injector location, it is essential to optimise calibration parameters such as injection pressure, injection and ignition timing. Furthermore, given that engine start is a highly transient process, it is necessary to adjust these parameters on a combustion event-by-event basis.

In the current context, ‘stable and reliable run-up’ is taken to mean consistency of the resulting engine speed response profile. In other words, after achieving a rapid first fire during cranking, the engine accelerates quickly and smoothly to the desired idle speed, without exhibiting aberrant behaviour, *e.g.* engine stall, excessive over or under-shoot, engine speed oscillations or engine stumble (non-smooth run-up). Ideally, at any temperature, the start profile should remain repeatable, regardless of the background variation or *noise*, *e.g.* fuel type, engine age or variations in fuel pressure. Hence, from a user-perspective, the objective is to discover a robust calibration, *i.e.* one which shows relatively low, or ideally no, variation to this background noise.

Whereas the control system provides compensators for some variables, *e.g.* start temperature, the effect of some operational properties is not directly accounted for, *e.g.* fuel type. In addition, some actuators cannot be consistently set, due to limitations of the control sub-system electro-mechanical response. For example, fuel pressure, which is ostensibly a controlled actuator, cannot be reliably set in practice and varies considerably from start-to-start. This limitation relates to the ability of the camshaft-driven mechanical high pressure fuel pump to deliver sufficient flow rate to charge the fuel rail at very low engine speed. As the instantaneous fuel delivered is directly related to the differential pressure across the injector, this represents a serious concern.

For noise variables of this type, there is no direct feed forward or feedback compensation available. Consequently, techniques have been borrowed from the principles of *parameter design* (Davis and Grove, 1992; Taguchi, 1993; Fowlkes and Creveling, 1995; Kawaguchi *et al.*, 2009) and minimising structural design sensitivity (Haug *et al.*, 1986; Cho and Jung, 2003) in an attempt to induce the necessary level

of robustness. As will be demonstrated comprehensively in this case study, robustness measures can conflict with their performance counterparts. This justifies the application of multi-objective optimisation approaches.

Furthermore, the inclusion of robustness measures in the optimisation process significantly increases the number of objectives. To reduce the dimensionality in this case study to a manageable level, engineering knowledge is applied *a priori* to limit the focus to robustness measures associated with the dominant noise variable, which is considered to be fuel pressure. Nevertheless, even with this simplification a constrained ten-objective optimisation problem results.

This chapter extends the application of the complexity reduction strategy, proposed in Chapter 4, to a high-dimensional real-world study, comprising ten objectives and one constraint. Four important enhancements are included and discussed in detail. These are:

1. **The Use of Parallel Computing Methods.** The computational demands on the process under investigation are now sufficiently high to justify a parallel computing approach. A parallel MOEA has been developed to evaluate large populations distributed across a cluster of processors. Batch processing in parallel has also been utilised to accelerate the clustering task.
2. **The Use of Concise Mathematical Notations.** These are introduced for reasons of clarity, brevity and efficiency for the clustering and objective reduction processes. Also, such notation lends itself to being implemented in software to automate these processes and in so doing, minimise errors.
3. **The Introduction of Sensitivity Objectives.** These have been added to the problem formulation so that the optimiser simultaneously searches for solutions, which are optimal for performance and which minimise the sensitivity to background noise.
4. **Variations on Thresholds for reducing the Number of Objectives.** Varying the threshold used for selecting Principal Components may affect the number of objectives retained using the objective reduction rules. This can provide flexibility in the dimension reduction process.

However, fundamentally the concept of *local harmony* is exploited to allow various degrees of complexity reduction in several local domains of the Pareto-Optimal Population (POP). The resultant sequence of optimisations, clustering and objective reduction processes enables the decision maker, working in conjunction with an experienced calibration engineer, to propose potential solutions. These results, developed systematically using the methods described, are shown to out-perform the existing calibration developed using empirical approaches.

### 6.1.1 Cold Start Profile Descriptions

Figures 6.1 and 6.2 show Net Mean Effective Pressure (NMEP) (Ferguson, 1986) and engine speed data for a ‘good’ and a ‘bad’ start respectively. NMEP is related to the chemical energy from combustion that is available at the crankshaft. Clearly, at any instant and for a given fuel quantity supplied, it is desirable to maximise this quantity. It can be seen from the NMEP traces, that the integrated NMEP for the good start is significantly greater than the bad start in the first twenty combustion events, due mostly to the *partial burns* (Heywood, 1988) of the bad start. In the corresponding engine speed traces, the good start exhibits a faster rise rate, higher peak flare speed and smooth decay to the desired idle speed. In contrast, the bad start shows a stumble in the first second, a slower rise rate, a lower flare speed and a less smooth decay with some evidence of undershoot.

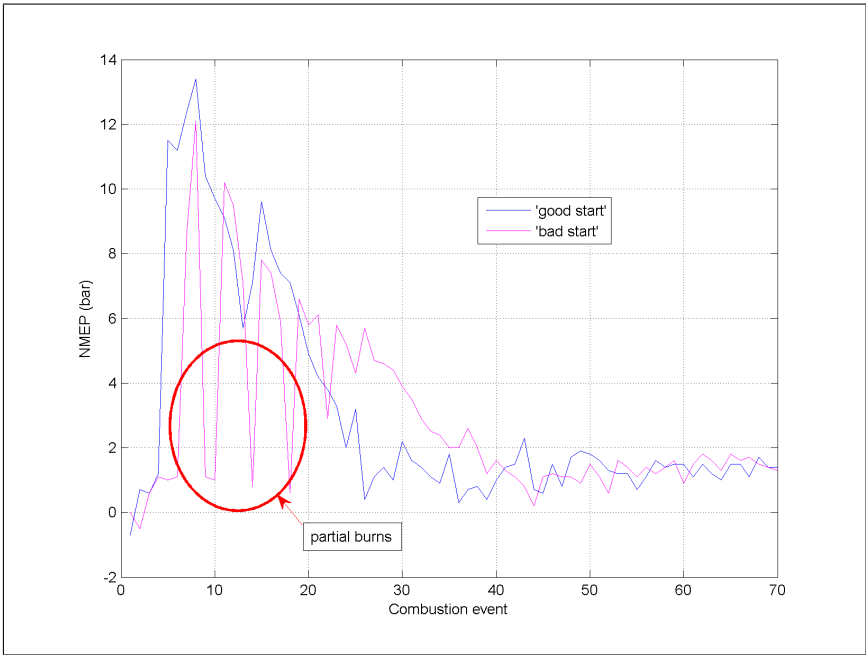


Figure 6.1: Cold start NMEP traces for a ‘good’ and a ‘bad’ start.

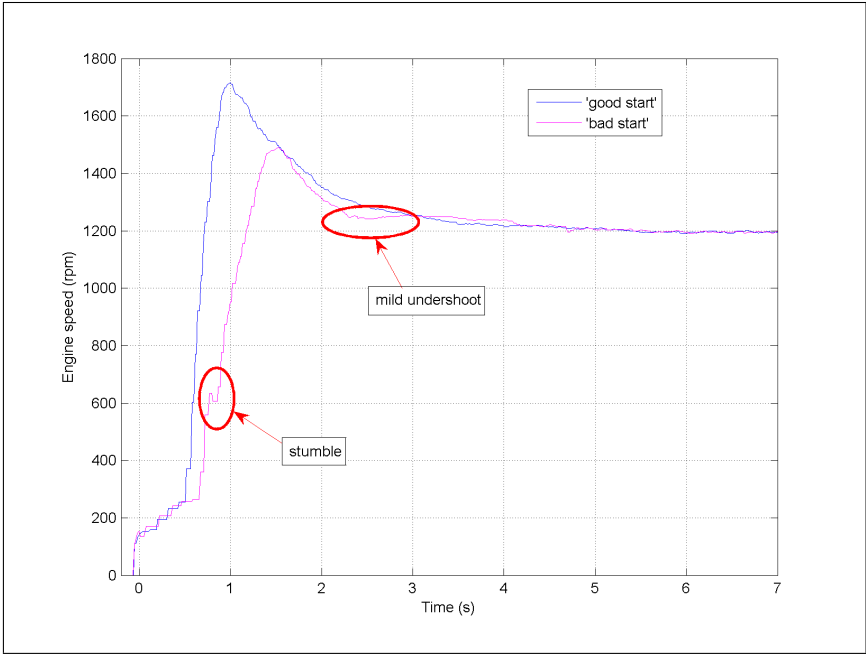


Figure 6.2: Cold start engine speed traces for a ‘good’ and a ‘bad’ start.

## 6.2 A Review of Sensitivity Analysis Approaches

As with many engineering problems, in engine calibration studies it is desirable to achieve a solution that is not only optimal in some sense, but also robust to variation. This background variation or *noise* comprises piece-to-piece variation in control system sensors and actuators, external environmental factors and customer duty (Davis and Grove, 1992). For the purposes of this case study, a robust, optimal calibration is defined as some optimal trade-off of competing engine responses, for which the solution is relatively insensitive to noise. In practice, it may not be possible to simultaneously achieve optimal performance and low sensitivity to noise and therefore some compromise may be necessary. This justifies a means by which the sensitivity of the response should be evaluated. There follows a brief review of sensitivity analysis approaches with their strengths and weaknesses. More extensive surveys can be found in Saltelli *et al.* (1999); Frey and Patil (2002); Helton and Davis (2003) and Tang *et al.* (2007).

Sensitivity analysis methods can be categorised (Frey and Patil, 2002) into Statistical or Mathematical methods as follows:

### 6.2.1 Statistical Sensitivity Analysis Approaches

These approaches involve assigned probability distributions to inputs to assess the effect on the output distribution. This category includes:

- **Monte Carlo Methods.** These comprise some probabilistic sampling of the inputs to develop a mapping of the input (or design) space to the output. Advantages are that extensive sampling is made of the input space and input interactions can be explored. The main drawback is the significant computational cost. In addition, it is not clear which probability distribution should be assumed for sampling the inputs and also, this technique cannot be incorporated directly into a multi-objective optimisation process. A further disadvantage is that it is a ‘Pass/Fail’ test in the sense that these methods only inform acceptable or unacceptable robustness unlike a sensitivity model, which may be explored by the decision maker to find acceptable sensitivity.
- **Fourier Amplitude Sensitivity Test (FAST).** In this approach, values of



each model input are transformed to values along a search curve. A frequency is specified for each input and using Fourier coefficients the output variance can be evaluated (Cukier *et al.*, 1973). Benefits of this method are that the full design space can be explored as well as the effect of input interactions. Among the disadvantages are the mathematical complexity, it is not well known or widely applied and importantly, can be computationally expensive.

The statistical methods described above are global in that they can consider the entire design space and take into account input interactions. However, this increased capability involves significant computational expense and prohibits their online usage in optimisation search processes.

*Signal-to-Noise (SN) ratios* are another statistical method, although not one involving the assignment of probability distributions to inputs. Taguchi (1987) introduced the concept of a SN metric to optimise the robustness of a product, where the signal represents the output or response of interest. The aim is to maximise the SN ratio to optimise the robustness. Much more explanatory detail with applications is provided in texts such as Fowlkes and Creveling (1995).

The main advantage of this approach is that there is only one SN ratio per objective, whereas with for example, direct derivatives, there is one sensitivity objective per input, so the number of sensitivity functions can increase quickly. However, the confounding of the signal and noise is a serious drawback (Nair, 1992). For example, a small signal equates to a small SN ratio even though the noise may be comparatively large and conversely, an extremely large signal would result in a large SN ratio, but the noise may still be large. In these scenarios maximising the SN ratio will not provide robust solutions.

### 6.2.2 Mathematical Sensitivity Analysis Approaches

These methods involve the determination of the change in output corresponding to a local change in the inputs. This category includes:

- **Controls Percentage Sensitivity.** This approach involves calculation of the fractional change in the output with respect to the fractional change in the input and can be expressed as a percentage (D'Azzo and Houpis, 1981). Benefits of this method are that i) it is unitless, ii) sensitivity objectives can be

easily created and traded-off against performance, iii) individual sensitivities are directly comparable and iv) it is useful for setpoint or process control, *e.g.* servo control. Conversely, a disadvantage is that the number of objectives are increased due to an additional objective per sensitivity. However, the most significant drawback is that if the target approaches zero, the sensitivity tends towards infinity. Since a target of zero can occur with some engine responses, *e.g.* driveline zero torque crossing, this approach is not sufficiently general for engine calibration problems and will not be considered further.

- **Direct Derivatives.** These methods are similar to those of the Controls Percentage Sensitivity approach, but are based on partial derivatives of the outputs with respect to changes in the inputs and therefore have units. However, if the inputs are first normalised, then all sensitivities are directly comparable.

Finite difference methods can be used to approximate partial derivatives based on small perturbations in the inputs and are straightforward to implement (Delinchant *et al.*, 2004; Tang *et al.*, 2007). Taking second order partial derivatives allows approximation of the rate of change of sensitivity or curvature in the vicinity of an optimum. As with the Controls Percentage Sensitivity approach, the number of objectives is increased as sensitivity objectives are added, but the Direct Derivatives approach does not suffer from the weakness of infinite sensitivity when the target performance is zero. Thus, this approach has all the advantages and addresses the disadvantages of the Controls Percentage Sensitivity method.

The primary aim for this case study is to include sensitivity and engine response objectives in the optimisation problem formulation so that both can be simultaneously searched for Pareto-optimal solutions. Requirements to achieve this aim comprise a computationally efficient, easy-to-implement approach, which can be integrated into the optimisation and is sufficiently general for engine calibration optimisation problems. The Direct Derivatives method most closely matches these requirements and is the selected sensitivity analysis approach for this case study.

### 6.3 Definition of Sensitivity Objective Functions

Applying a forward finite difference approximation to a partial derivative results in the following definition of a sensitivity objective function,  $S_m$  for the  $m^{th}$  objective, as used in this case study:

$$S_m(x_{inp}) = \frac{f_m(x_{inp} + \Delta x_{inp}) - f_m(x_{inp})}{\Delta x_{inp}} \quad (6.1)$$

where  $f_m$  is the  $m^{th}$  objective function,  $x_{inp}$  is the  $inp^{th}$  objective function input or decision variable and  $\Delta x_{inp}$  is a small perturbation applied to the decision variable to generate a change in the objective function and hence a sensitivity. The value used for  $\Delta x_{inp}$  was based on a relatively small percentage of the decision variable range. This was determined simply by evaluating the sensitivity objective functions at various values of  $\Delta x_{inp}$  against the decision variable under consideration (Fuel Pressure, FP), plots of which are shown in Figure 6.3. The value of  $\Delta x_{inp}$  was chosen to be 0.1%, which was considered sufficiently small to give an adequate approximation to the partial derivative.

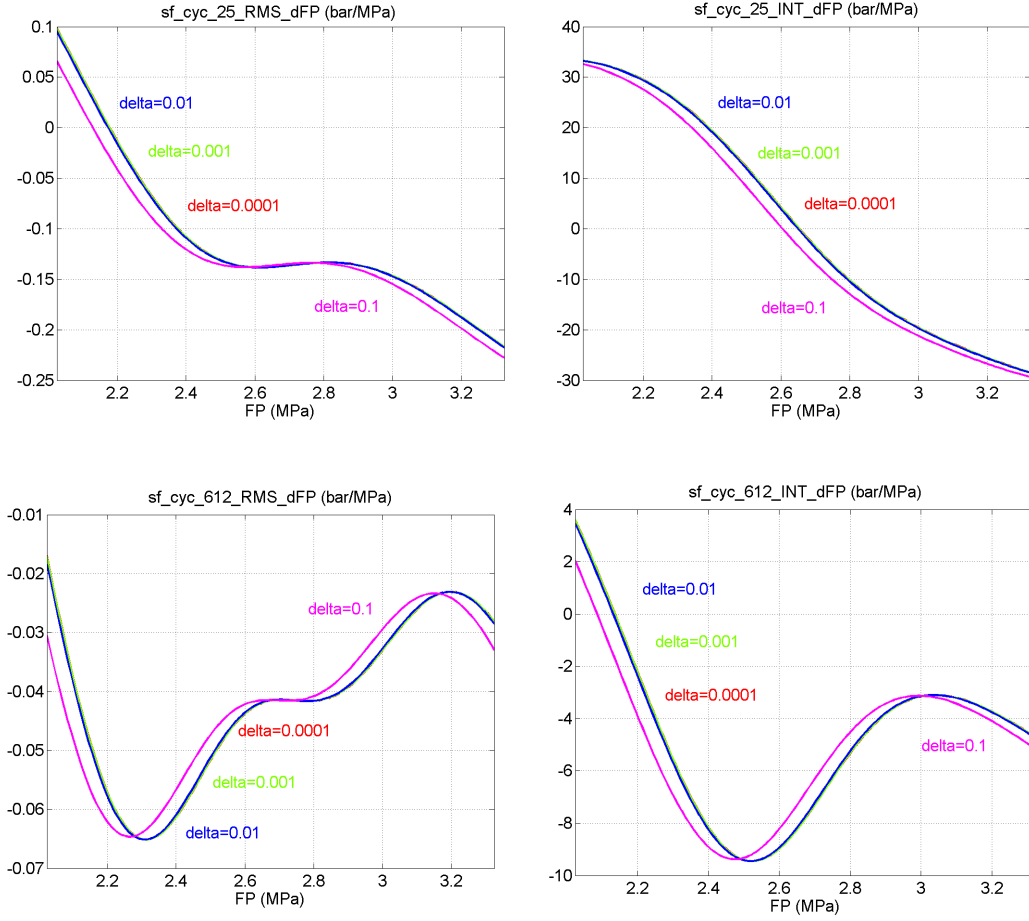


Figure 6.3: Sensitivity objective functions plotted against Fuel Pressure (FP) for various values of  $\Delta x_{inp}$  shown as proportions. The other decision variables were set at the centre of their range, except for End Of Injection (EOI), which was set to 75 degrees BTDC from a previous optimisation.  $\Delta x_{inp} = 0.001 = 0.1\%$  was selected to give an adequate approximation to the partial derivative. Y-axis labels are at the top of each plot.

## 6.4 pMOEA Implementation

As stated in Section 2.3.4, given that a compute cluster was available and the objective functions were fast to evaluate for this research, the only suitable pMOEA paradigm was the island-based approach. The implementation took the form of a parallelisation of the modified NSGAII of Section 4.3.3. The Matlab<sup>®</sup> v7.7 (R2008b) Parallel Computing Toolbox and the Distributed Computing Server were used to allow sub-populations to be evaluated in parallel, periodic migration between islands (processors) and collation of the final population.

The parallelisation was implemented by The MathWorks Limited at the request of

the author and comprised:

- Splitting the population up by the number of processors available on the compute cluster. This cluster comprised a total of ten office PCs with multi-core processors, a large proportion of which were not used during the course of normal day-to-day operations. During trials, it was found that the number of processors had to be limited to those on the same sub-network and those not being intensively used, which amounted to between thirty-five and forty processors.
- Specifying a number of individuals from the population to migrate between neighbouring processors together with a migration frequency. The number of immigrants used was 2% of the island population migrated every generation. These settings were consistent with those used, with some success, in Streichert *et al.* (2005).
- Immigrants were chosen by selecting pairs of individuals with each pair comprising the fittest and least fittest individuals. Whilst it may seem intuitive to choose only the fittest solutions to migrate, there is the possibility that in doing so, premature convergence could result. Including the worst individual(s) introduced some diversity and was straightforward to implement.

Clearly, there are opportunities to explore alternative migration parameter settings and schemes. Since the aforementioned parallel Matlab<sup>®</sup> software was only available for a limited period on trial licenses, such studies have been left for future research. A validation test was conducted to verify that the implemented pMOEA generated broadly equivalent Pareto-Optimal Populations (POPs). This test was based on the six objective diesel case study of Chapter 5. As can be seen from the pairwise objective scatter plots in Figure 6.4, when overlaid, the Pareto-Optimal Populations resulting from runs with the serial and parallel versions of NSGAII corresponded well in terms of objective pattern and distribution.

In addition, it was of interest to compare the execution times of the serial and parallel algorithms. With all parameters the same, the serial algorithm on a single processor took 1271 minutes, whereas the parallel algorithm using twenty equivalent processors took only fifteen minutes - a considerable time saving.

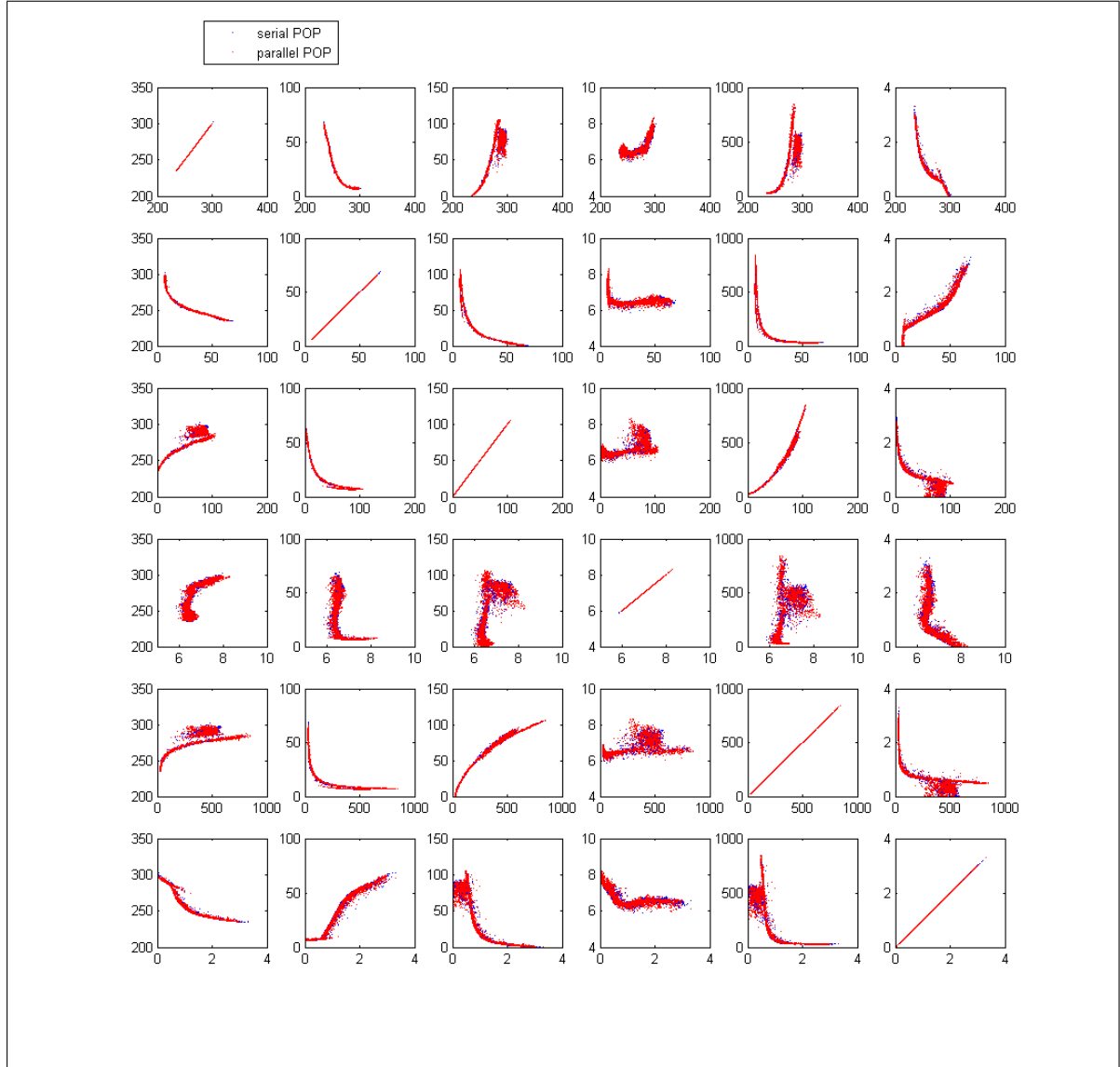


Figure 6.4: Pareto-Optimal Populations (POPs) resulting from runs with serial and parallel versions of NSGAI. Objective axes are not labelled as they are not required and to make the plot more compact.

A study was conducted into run time for the six-objective diesel problem with a varying number of processors (Emtage, 2009). In doing so, it was discovered that part of the speed-up can be attributed to internal operation of the CPU; specifically, the size (32kB) of the L1 cache on Intel® Core™ 2 Quad CPU Q9450 2.66 GHz processors. With less than twelve processors, the memory taken by the objective and decision variable matrices exceeded 32kB, in which case the slower L2 cache would be used. Conversely, with more than twelve processors, only the L1 cache was used and

the speed-up was clearly improved. Consequently, it was concluded that the implemented pMOEA would be adequate for use in this many-objective cold start optimisation problem.

## 6.5 Problem Formulation

The objective functions used in the cold start case study were based on empirical engine models. These were developed from experimental data taken on a 2 litre in-line four cylinder turbocharged direct injection gasoline passenger car engine. The data comprised a series of cold start tests each from an initial engine coolant temperature of 20 °C. After each start, the engine was fully warmed-up to a stabilised temperature to burn off any residual hydrocarbon emissions and fuel in the oil (Heywood, 1988). The engine was then switched off and chilled back down to the initial coolant temperature in preparation for the next start. Fifty-seven starts were conducted as part of a designed experiment with ten validation tests.



Figure 6.5: Encapsulated dynamic dynamometer cell 109 in the West Wing Laboratory at the Ford Dunton Technical Centre, Essex.

The engine test facility used was a dynamic dynamometer encapsulated test cell, a photograph of which is shown in Figure 6.5. Such facilities can be provide very efficient, cost effective and realistic testing on a rig as opposed to building expensive prototype vehicles, which require specialised vehicle-based test facilities or testing in remote cold climate locations.

### 6.5.1 Combustion Quality Metrics

In addition to the inclusion of sensitivity functions in the optimisation, several combustion quality metrics were incorporated as follows:

- **Dynamic Measures of Combustion Intensity.** Cumulative Net Mean Effective Pressure (NMEP), summed over several engine cycles is used as a measure of localised combustion intensity and is defined as:

$$J_a^b = \sum_{w=s+(a-1)c}^{s+bc-1} x_w, \quad \forall x_w \in W \quad (6.2)$$

where  $J$  is the combustion intensity for cycles  $a$  to  $b$ ,  $c$  is the number of engine cylinders,  $s$  is the combustion event index of the first fire during engine crank and  $x_w$  is the NMEP for a combustion event in an ordered dataset,  $W$ , of all combustion events from first fire onwards. For example, the combustion intensity metrics used for this case study are  $J_2^5$  and  $J_6^{12}$  and assuming  $c = 4$ :

$$J_2^5 = \sum_{s+4}^{s+19} x_w \quad (6.3)$$

and,

$$J_6^{12} = \sum_{s+20}^{s+47} x_w \quad (6.4)$$

Both  $J_2^5$  and  $J_6^{12}$  are measures of the cumulative energy supplied by combustion over the cycles selected. Maximising these characteristics is equivalent to



maximising the energy available to accelerate the engine rotational speed after start, known as the *run-up* period.

- **Dynamic Measures of Combustion Variation.** In steady-state combustion development studies, standard deviation or coefficient of variation have been used as combustion metrics (Ferguson, 1986; Heywood, 1988; Brunt and Emtage, 1996). During steady-state operation, these can be calculated over a relatively large number of cycles and on a lumped engine or individual cylinder basis. This is straightforward because the mean NMEP response, for specific or all cylinders, is essentially constant throughout. However, by definition it is not possible to use such metrics in dynamic studies.

In this study a dynamic combustion variation measure is proposed, based on the root-mean-square (RMS) of the instantaneous combustion deviations. The set of squared NMEP deviations,  $DN$ , is defined in Equation C.4:

$$DN = \{x_w, y_w \in \mathbb{R} \wedge d_w^2 = (y_w - x_w)^2\}_{w=s}^N \quad (6.5)$$

where  $y_w$  is the Savitzky-Golay (Savitzky and Golay, 1964) smoothed NMEP data (see Appendix C for further detail), such that any deviations between the smoothed and raw NMEP data indicate the quality or variation in the combustion process. This approach is somewhat similar to methods in statistical time series analysis (Box and Jenkins, 1970; Brockwell and Davis, 1991) where the filtered response is subtracted from the raw data to de-trend the data prior to estimating the cyclic correlation trends.

If  $R_a^b$  denotes the RMS of the instantaneous NMEP values over cycles  $[a, b]$ , then:

$$R_a^b = \sqrt{\frac{\sum_{w=s+(a-1)c}^{w=s+bc-1} d_w^2}{(1 + (b - a))c}}, \quad \forall d_w \in DN \quad (6.6)$$

Based on the discussion in Appendix C,  $R_a^b$  has the disadvantage that the magnitude of the individual deviations depends on the filter parameters. This property significantly devalues this statistic as a generic dynamic combustion quality indicator. However, in a specific case-study utilising a SG-filter with fixed parameters, minimisation of this quantity is still meaningful. Again, assuming  $c = 4$ , the appropriate metrics are:

$$R_2^5 = \sqrt{\frac{\sum_{w=s+4}^{w=s+19} d_w^2}{16}} \quad (6.7)$$

and,

$$R_6^{12} = \sqrt{\frac{\sum_{w=s+20}^{w=s+47} d_w^2}{28}} \quad (6.8)$$

### 6.5.2 Optimisation Problem

The optimisation was formulated as a ten-objective, single constraint problem as follows:

Minimise:

- Combustion variation metric for cycles 2-5, cyc\_25\_RMS (bar).
- Combustion variation metric for cycles 6-12, cyc\_612\_RMS (bar).
- Negative run-up combustion intensity for cycles 2-5, neg\_25\_INT (bar).
- Negative run-up combustion intensity for cycles 6-12, neg\_612\_INT (bar).
- Fuel quantity, F\_model (unitless). Due to measurement difficulties, this was used as a surrogate measure for HC emissions, which varies in proportion to the fuel quantity injected.
- Maximum engine speed flare after start, Peak\_Flare\_Speed, (rpm).
- Absolute value of sensitivity of combustion variation metric for cycles 2-5 to Fuel Pressure, sf\_cyc\_25\_RMS\_dFP\_abs (bar/MPa).

- Absolute value of sensitivity of combustion variation metric for cycles 6-12 to Fuel Pressure, `sf_cyc_612_RMS_dFP_abs` (bar/MPa).
- Absolute value of sensitivity of run-up combustion intensity for cycles 2-5 to Fuel Pressure, `sf_cyc_25_INT_dFP_abs` (bar/MPa).
- Absolute value of sensitivity of run-up combustion intensity for cycles 6-12 to Fuel Pressure, `sf_cyc_612_INT_dFP_abs` (bar/MPa).

Subject to:

- A constraint on the mild extrapolation of valid domain or boundary of the models. This model boundary is an envelope wrapped around the boundary of the data used to build the models and allows, in this case, mild extrapolation (MathWorks, 2008a). The constraint is defined as model boundary  $\leq 0.15$ .

All models had the following inputs, all of which were used as decision variables:

- AIR - inducted air mass flow (kg/h) as controlled by the engine throttle.
- DEC - exponential decay (unitless) in injected fuel quantity.
- SPK2 - crankshaft angle timing (degrees before piston top dead centre) of ignition.
- F - injected fuel quantity, expressed as a factor (unitless).
- FP - fuel pressure (MPa). Limited control on this control system actuator is available during cold start operation. Therefore, there is a requirement to develop a cold start calibration that is not only optimal for cold start performance, but also relatively insensitive to variation in fuel pressure. Hence, FP was the decision variable under consideration in the formulation of the sensitivity objective functions.
- EOI - crankshaft angle timing (degrees before piston top dead centre) of end of fuel injection. This was fixed at a value of 75 degrees BTDC from a previous optimisation.

(It should be noted that this problem contains 5 decision variables and 10 objectives so is underdetermined.) The pMOEA described in Section 6.4 was used with the following options and parameters specified:

- Number of generations: 50000.
- Population size: 20000 - both of these first two parameters were chosen to provide a reasonable computational effort compared to the diesel six-objective case study, where an initial population of 4000 was run for 5000 generations; in addition, as with the six-objective case study, a reasonably large initial population is used to allow cluster-based sub-sampling consistent with cluster verification rule CR2.
- External archive used and updated every 1 generation.
- Number of migrants: 2% of island population.
- Migration frequency: every 1 generation.
- Selection operator: tournament of size 2.
- Crossover operator: SBX (Deb and Agrawal, 1995) with probability: 0.7 and distribution index: 20.
- Mutation operator: polynomial (Deb and Goyal, 1996) with probability: 0.17 and distribution index: 20 (Deb, Thiele, Laumanns and Zitzler, 2002; Khare *et al.*, 2003).

Table 6.1: Decision variable ranges.

Decision Variable	Units	Ranges	
		Minimum	Maximum
AIR	°BTDC	25	45
DEC		0	0.104
SPK2		-10	10
F		1.49	3.49
FP	MPa	2.02	3.5

- Minimum and maximum range limits on the decision variables are listed in Table 6.1. These were not explicitly included in the problem formulation. Instead, a so-called *boundary constraint model* (referred to as model\_bdry) was

incorporated, which represents a convex hull envelope around the data. Beyond this `model_bdry`, models built from the data are extrapolated.

- Initially, all the objectives were minimised subject to the constraint, which corresponds to the *constrained minimisation* formulation in the Progressive Preference Articulation method of Fonseca and Fleming (1998a) ( $PPA_{FF}$ ) as described in Section 4.7.1. The resulting initial goals and priorities are shown in Table 6.2.

Results from the optimisation process are presented in the next section. This process comprises four stages of successive objective reductions. The analysis involved is summarised in flowcharts with accompanying explanations. Details of the analysis for each of the four stages are provided in Appendices: F to I, respectively.

Table 6.2: Initial goals and priorities for the ten-objective cold start problem, where the last row represents the constraint.

Objective	Goal	Priority
neg_25_INT	$-\infty$	1
neg_612_INT	$-\infty$	1
cyc_25_RMS	0	1
cyc_612_RMS	0	1
F_model	0	1
Peak_Flare_Speed	0	1
sf_cyc_25_RMS_dFP_abs	0	1
sf_cyc_612_RMS_dFP_abs	0	1
sf_cyc_25_INT_dFP_abs	0	1
sf_cyc_612_INT_dFP_abs	0	1
model_bdry	0.15	2

## 6.6 Results from the Objection Reduction Process - 1st Stage

A summary of the 1st stage objective reduction process after the initial optimisation is provided in the flowchart in Figure 6.6. The resulting Pareto-Optimal Population

(POP) of 18,552 solutions is robustly clustered, suitable sub-sample clusters determined and then principal component analysis (PCA) applied to potentially reduce the number of objectives within each cluster. Mathematical notations are used in the clustering and objective reduction blocks, which will be described in the following sections.

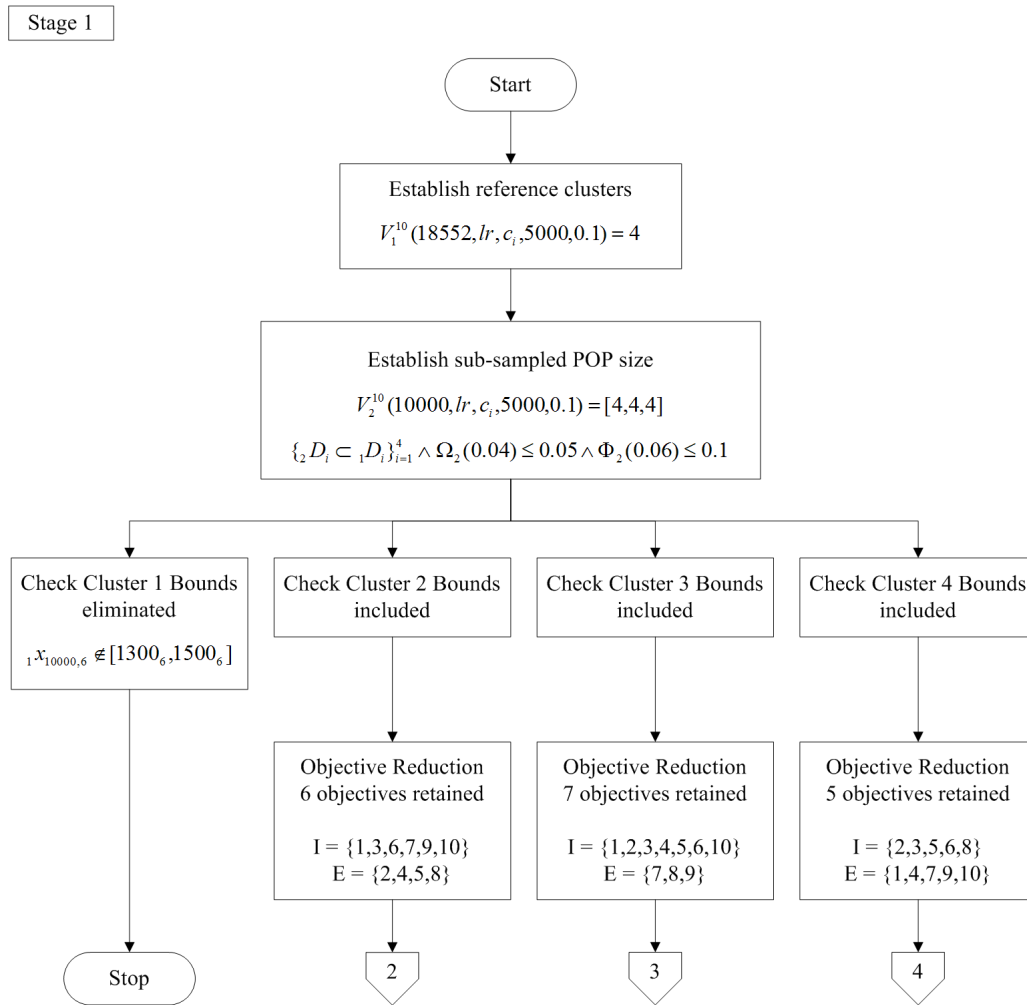


Figure 6.6: Flowchart of the results from objective reduction - 1st stage, leading to Clusters 2, 3 and 4 with 6, 7 and 5 objectives being retained, respectively. This flowchart starts after the first optimisation has been completed.

### 6.6.1 Clustering and Verification

In order to carry out the clustering, the Clustering verification Rules as defined in Section 4.4.3 were applied. However, for high-dimensional optimisation problems, the application of these rules requires lengthy description. To address this, a compact mathematical notation has been devised and is defined in Appendix D. Such a notation lends itself to being coded in software to automate the application of these rules to make their usage much more efficient.

The following summarises the sequence of steps depicted in the flowchart in Figure 6.6 with corresponding details provided in Appendix F:

- **Establish Reference Clusters.** The first step was to cluster the *reference POP* generated from the initial optimisation. The results can be summarised using the notation,  $V_1^{10}(18552, lr, cs, 5000, 0.1) = 4$ . That is, the reference clustering analysis in ten objectives, of the reference POP of 18,552, from various learning rates,  $lr$ , and initial number of clusters,  $cs$ , maximum iterations of 5000 and a convergence tolerance of 0.1 generated a *reference solution* of four converged clusters.
- **Establish Sub-sampled POP Size.** Subsequently, the reference solution was randomly sub-sampled per cluster to generate smaller POPs of 10000, 5000, 2000 and 1000. Clustering was run on all of the sub-sampled POPs to test for agreement with the reference solution clusters. As a result, it was decided that the POP of 10,000 was the smallest sub-sampled POP that provided acceptable agreement with the reference POP. This is denoted by:
  - $V_2^{10}(10000, lr, cs, 5000, 0.1) = [4, 4, 4]$  - a clustering analysis on a sub-sampled POP of 10,000 with various  $lr$  and  $cs$  resulted in three alternative solutions of four clusters. A run with the best convergence was selected from those in the first (most frequently occurring) solution.
  - $\{ {}_2D_i \subset {}_1D_i \}_{i=1}^4 \wedge \Omega_2(0.04) \leq 0.05 \wedge \Phi_2(0.06) \leq 0.1$  - with respect to the corresponding reference clusters, the selected sub-sampled 10,000 POP clusters are a subset AND have cluster centres in close agreement AND have cluster correlation matrices in close agreement.
- **Check Cluster Bounds.** The only engineering limit that was specified by the

Cold Start calibration engineer, was applied to Objective 6 (Peak\_Flare\_Speed), where only solutions in the 1300 to 1500rpm range were of interest. The Peak\_Flare\_Speed data within Cluster 1 violated this limit and so was discarded, while the other clusters satisfied this limit and were retained. The cluster data bounds per objective are detailed in Table F.4 in Section F.1.3.

### 6.6.2 PCA and Potential Objective Reduction

As with the six-objective problem in the previous chapter, the objective reduction rules (see Section 4.5) were applied to each cluster to identify any opportunity for potential objective reduction with the results summarised in Table F.5. In the process of discussing and selecting clusters to be retained in Section 6.6.1, the Cold Start calibration engineer advised objective priorities as shown in Table 6.3 and these were taken into account, where necessary, when applying the objective reduction rules. The notation defined in Appendix E has been used to more concisely represent the objective reduction process. An example is given below of how this notation is applied with a brief description provided for clarification. The remaining application of these rules is provided in Section F.2.

Table 6.3: Initial goals and priorities for the ten-objective cold start problem, where the last row represents the constraint.

Objective	Goal	Priority
neg_25_INT	$-\infty$	4
neg_612_INT	$-\infty$	4
cyc_25_RMS	0	2
cyc_612_RMS	0	2
F_model	0	6
Peak_Flare_Speed	0	5
sf_cyc_25_RMS_dFP_abs	0	1
sf_cyc_612_RMS_dFP_abs	0	1
sf_cyc_25_INT_dFP_abs	0	3
sf_cyc_612_INT_dFP_abs	0	3
model_bdry	0.15	7



### Retaining PCs to account for 95% of the Total Variation

#### Objective Reduction Rules applied to Cluster 2

1. Applying OR1,  $\lambda_1/\lambda_{10} = 164.5$ , so there was insufficient evidence of a near-linear dependency.
2. Applying OR2,  $t_5 = 95\%$ , *i.e.* first five PCs were retained, which account for approximately 95% of the cumulative percentage total variation.
3. Applying OR3a),  $\theta_{10} = 0.3162$  - the eigenvectors for the five retained PCs were rotated and the threshold for the test of significance was  $10^{-0.5} = 0.3162$ .
4. Applying OR3c) to PC1,  ${}_1A_-(2, 4, 5) \wedge {}_1A_+(8) \wedge (r_5 > r_2 > r_4) \Rightarrow \{5, 8\} \subseteq I$  - the rotated eigenvector coefficients,  $v_2^1$ ,  $v_4^1$ ,  $v_5^1$  and  $v_8^1$  exceeded 0.3162 in magnitude and thus were classified as significant.  $v_2^1$ ,  $v_4^1$  and  $v_5^1$  have the same sign, but Objective 5 (F\_model) had the highest priority and so was retained.  $v_8^1$  was the only positive significant coefficient and so Objective 8 (sf\_cyc\_612\_RMS\_abs) was also retained.
5. Applying OR3c) to PC2,  ${}_2A_-(6) \wedge {}_2A_+(8) \Rightarrow \{6, 8\} \subseteq I$  -  $v_6^2$  and  $v_8^2$  were the only two significant coefficients and so both Objectives 6 (Peak\_Flare\_Speed) and 8 (sf\_cyc\_612\_RMS\_abs) were retained.
6. Likewise applying OR3c) to PC3,  ${}_3A_-(9) \wedge {}_3A_+(10) \Rightarrow \{9, 10\} \subseteq I$  -  $v_9^3$  and  $v_{10}^3$  were the only two significant coefficients and so both Objectives 9 (sf\_cyc\_25\_INT) and 10 (sf\_cyc\_612\_INT) were retained.
7. Applying OR3b) to PC4,  ${}_4A_+(1, 3) \wedge (r_1 > r_3) \Rightarrow \{1\} \subseteq I$  - the only significant coefficients were  $v_1^4$  and  $v_3^4$ , but Objective 1 (neg\_25\_INT) had the highest priority and so was retained.
8. Applying OR3b) to PC5,  ${}_5A_+(7) \Rightarrow \{7\} \subseteq I$  - the only significant coefficient was  $v_7^5$  and so Objective 7 (sf\_cyc\_25\_RMS\_abs) was retained.

In summary, seven objectives were retained in Cluster 2:

- F\_model
- sf\_cyc\_612\_RMS\_abs

- Peak\_Flare\_Speed
- sf\_cyc\_25\_INT\_abs
- sf\_cyc\_612\_INT\_abs
- neg\_25\_INT
- sf\_cyc\_25\_RMS\_abs

Overall, some objective reduction was achieved in Clusters 2 and 4 by retaining only those PCs that accounted for approximately 95% of the variation. However, it is of interest to understand what effect a reduced percentage of variation would have on objective reduction.

#### The Effect of a Reduced Percentage of Variation on Objective Reduction

Consequently, it was decided to consider two further such scenarios where PCs are retained that account for approximately 90% and 86% of the variation. The details of the remaining application of the objective reduction rules are given in Section F.2 and summarised in Table F.5. The results from the objective reduction process for varying thresholds of cumulative percentage of variation are collated in Table 6.4. It can be seen that while some objective reduction was achieved using a threshold of 95%, using 86% gives significantly more reduction. As a result, it was decided to proceed with retaining the objectives per cluster corresponding to the 86% threshold in further optimisation. This stage culminated in three Clusters (2, 3 and 4) with six, seven and five objectives being retained, respectively.

Table 6.4: 1st Stage retained objectives from objective reduction process for varying thresholds of cumulative percentage of total variation.

Threshold for cumulative % of total variation	No. of objectives retained		
	Cluster 2	Cluster 3	Cluster 4
95	7	10	7
90	6	8	7
86	6	7	5

## 6.7 Results from the Objection Reduction Process - 2nd Stage

### 6.7.1 Optimisation with Reduced Objectives - 2nd Stage

Three subsequent optimisations with the reduced objectives were carried out, one in each of the retained clusters. Previous optimisation parameter settings were used as defined in Section 6.5.2 with the exception of the population size, which is now 10,000 (was 18,552) as a result of sub-sampling (see Section 6.6.1) and the number of generations, which is now 25,000 (was 50,000) as the number of objectives and population size has significantly reduced, but is still considered a reasonable computational effort. The objective goals and priority order have been inherited from the previous objective reduction process (see Table 6.3) and are shown in Table 6.5, where the ellipsoidal cluster boundary model has been added as a constraint in an attempt to preserve objective correlations within the cluster boundary. The initial population used was that of the sub-sample, some of which would initially be

Table 6.5: Retained objectives from objective reduction process with associated goals and priorities consistent with those specified previously in Table 6.3, where Obj1 and Con1 refer to Objective 1 and constraint 1 respectively.

Objective/Constraint					
No.	Name	Goals	Priorities		
			Cluster 2	Cluster 3	Cluster 4
Obj1	neg_25.INT	$-\infty$	4	3	0
Obj2	neg_612.INT	$-\infty$	0	3	3
Obj3	cyc_25.RMS	0	2	1	2
Obj4	cyc_612.RMS	0	0	1	0
Obj5	F_model	0	0	5	5
Obj6	Peak_Flare_Speed	0	5	4	4
Obj7	sf_cyc_25.RMS_abs	0	1	0	0
Obj8	sf_cyc_612.RMS_abs	0	0	0	1
Obj9	sf_cyc_25.INT_abs	0	3	0	0
Obj10	sf_cyc_612.INT_abs	0	3	2	0
Con1	model_bdry	0	6	6	6
Con2	cluster_bdry	0	6	6	6

infeasible according to the cluster boundary constraint. This was not expected to be a problem, as these solutions would quickly be replaced by feasible ones as the algorithm proceeded.

A summary of the 2nd stage objective reduction process after the optimisation is provided in the flowchart in Figure 6.7. This stage culminated in four Clusters (2, 3, 4.1 and 4.2) with four, five, four and four objectives being retained, respectively. The next two sections summarise the sequence of steps with corresponding details

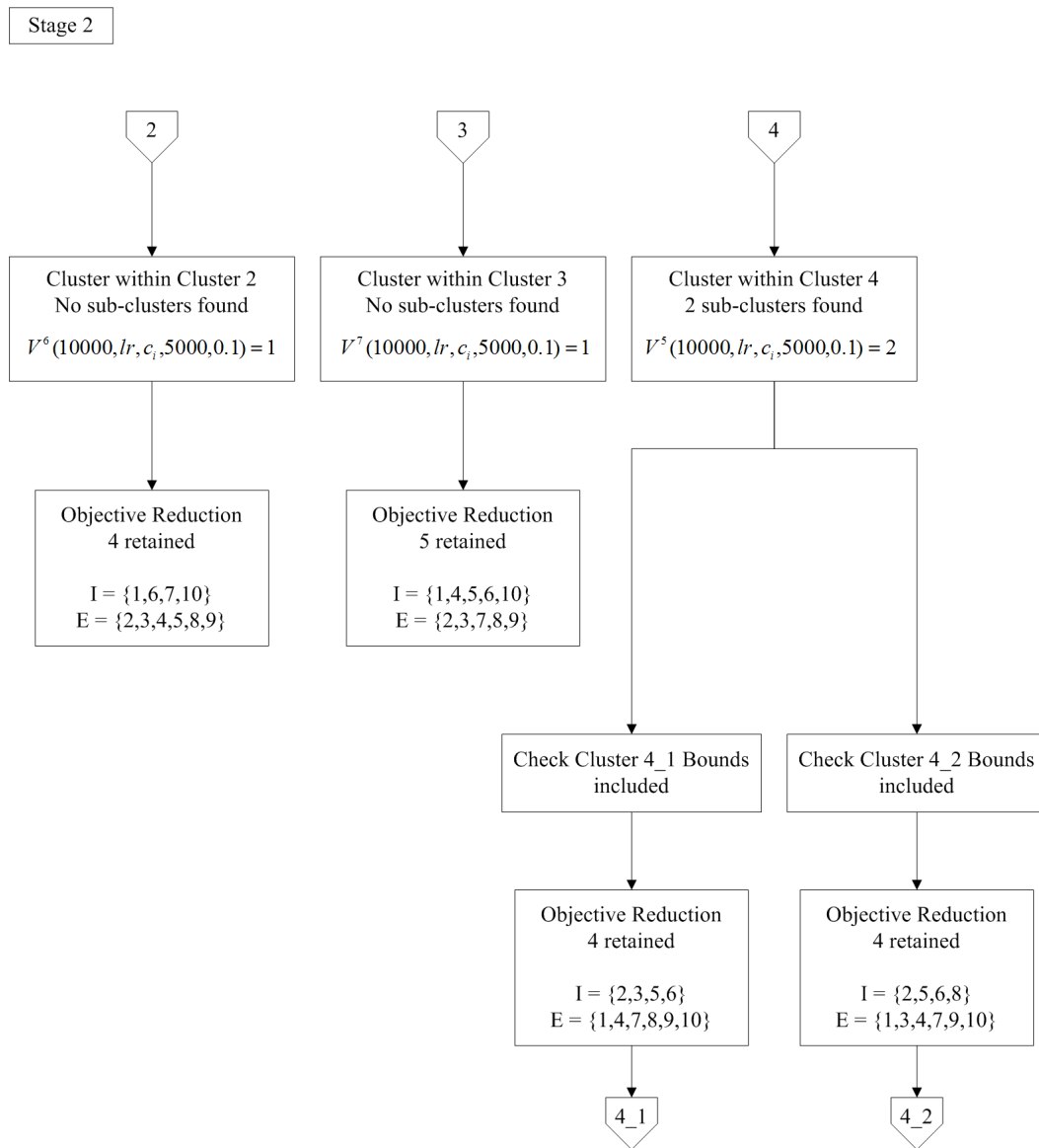


Figure 6.7: Flowchart of the results from objective reduction - 2nd stage, leading to Clusters 2, 3, 4.1 and 4.2 with 4, 5, 4 and 4 objectives being retained, respectively. This flowchart starts after the optimisation in the 2nd stage has been completed.

provided in Appendix G.

### 6.7.2 Clustering and Verification with Reduced Objectives - 2nd Stage

The final populations resulting from the Optimisation with reduced objectives - 2nd Stage were clustered using the k\*-Means clustering algorithm and initial parameters set as previously in Section 6.6.1. Further detail is provided in Section G.1.

- **Cluster within Clusters.** Initial clustering runs on each of the populations did not produce any sub-clusters. It was found that, although the data within each population was unique using the default Matlab<sup>®</sup> double precision of 16 decimal places, when a user-specified engineering precision of 1 to 3 decimal places was used far fewer unique solutions existed. Consequently, it was decided to reformulate the objective functions to engineering precision in an attempt to generate more unique (to engineering precision) solutions (see Section G.1 for more detail). The final populations from the optimisations of the re-formulated objectives were then clustered. This resulted in no sub-clusters in the retained Clusters 2 and 3 and two sub-clusters in retained Cluster 4, *i.e.*

$$\text{For Cluster 2: } V^6(10000, lr, cs, 5000, 0.1) = 1 \quad (6.9)$$

$$\text{For Cluster 3: } V^7(10000, lr, cs, 5000, 0.1) = 1 \quad (6.10)$$

$$\text{For Cluster 4: } V^5(10000, lr, cs, 5000, 0.1) = 2 \quad (6.11)$$

As the computational expense of optimising a population of 10,000 over 10 objectives was considered acceptable with the pMOEA running on more than 30 processors, (2-3 hours), further sub-sampling was not carried out (*i.e.*  $V$  has no suffix). Consequently, Clustering verification Rules, CR2 and CR3 did not apply.

- **Check Sub-Cluster Bounds.** Applying CR4, no further engineering limits were applied by the Cold Start calibration engineer to the sub-clusters in Cluster 4. In addition, both clusters had significant membership and so were retained. The bounds on the cluster data are shown in Table G.2 in Section G.1.

### 6.7.3 PCA and Potential Objective Reduction - 2nd Stage

For the final populations from the optimisations of the re-formulated objectives in Clusters 2 and 3, FAST-MCD was used to determine robust correlation matrices, on which to base PCA and as before, the proportion of the population retained was 95%. Subsequently, PCA was conducted and the results are shown in Table G.3 in Section G.2. The grey-shaded eigenvector coefficients indicate the retained objectives resulting from application of the objective reduction rules.

A similar process to that in Section 6.6.2 was used to explore what effect varying the cumulative percentage of total variation had on the number of objectives retained. In order to achieve this the Objective reduction Rules OR3a)-3c) were applied, the results of which are shown in Table G.4. The details of how these objective rules were applied is provided in Section G.2.

A reasonable objective reduction was achieved in Cluster 3 with a 99% threshold for the cumulative percentage of variation, but no objective reduction was achieved for Clusters: 2, 4\_1 and 4\_2. With a 95% threshold, there was a small objective reduction in Cluster 2, but none in Clusters: 4\_1 and 4\_2. With an 85% threshold, a reasonable objective reduction was achieved for Clusters 2, 4\_1 and 4\_2. However, for Cluster 3, an 85% threshold was considered to give too drastic an objective reduction in one go and so, it was decided to adhere to using a 99% threshold for this cluster. For this case study a policy of progressively, gradually reducing the number of objectives was adopted, albeit that this approach may involve more objective reduction stages and

Table 6.6: 2nd Stage retained objectives from objective reduction process for varying thresholds of cumulative percentage of total variation. The objective reduction selected per cluster is shown highlighted in grey. The number of retained objectives from the first stage is shown in brackets for each cluster.

Threshold for cumulative % of total variation	No. of objectives retained			
	Cluster 2 (6)	Cluster 3 (7)	Cluster 4_1 (5)	Cluster 4_2 (5)
99	6	5	5	5
95	5	4	5	5
85	4	3	4	4

therefore take longer. In summary:

- **Cluster 2.** Four objectives were retained: neg\_25\_INT, Peak\_Flare\_Speed, sf\_cyc\_25\_INT\_abs and sf\_cyc\_612\_INT\_abs.
- **Cluster 3.** Five objectives were retained: neg\_25\_INT, cyc\_612\_RMS, F\_model, Peak\_Flare\_Speed and sf\_cyc\_612\_INT\_abs.
- **Cluster 4\_1.** Four objectives were retained: neg\_612\_INT, cyc\_25\_RMS, F\_model and Peak\_Flare\_Speed.
- **Cluster 4\_2.** Four objectives were retained: cyc\_25\_RMS, F\_model, Peak\_Flare\_Speed and sf\_cyc\_612\_RMS\_abs.

## 6.8 Results from Objection Reduction Process - 3rd Stage

### 6.8.1 Further Optimisation with Reduced Objectives - 3rd Stage

Four subsequent optimisations were carried out with the reduced objectives from Stage 2, one in each of the retained clusters. Previous optimisation parameter settings were used as defined in Section 6.7.1. The objective goals and priority order have been inherited from the previous objective reduction process (see Table 6.5) and are shown in Table 6.7, where an additional ellipsoidal cluster boundary model has been added as a constraint for the two clusters resulting from Cluster 4.

A summary of the 3rd stage objective reduction process after the optimisation is provided in the flowchart in Figure 6.8.

### 6.8.2 Clustering and Verification with Reduced Objectives - 3rd Stage

The final populations resulting from further optimisation with reduced objectives - 3rd Stage, were clustered using the k\*-Means clustering algorithm. The initial parameters were set as previously in Section 6.6.1, except for the number of runs per cluster which was set to 19 (78 in total) and the learning rate which was set to a range of 1e-2 to 1e-5. This is a larger learning rate than used previously, but is consistent with the reduced number of objectives, as there are fewer parameters (smaller means

Table 6.7: Retained objectives from objective reduction process with associated goals and priorities consistent with those specified previously in Table 6.3, where Obj1, Con1 and n/a refer to Objective 1, constraint 1 and not applicable respectively.

Objective/Constraint						
No.	Name	Goals	Priorities			
			Cluster 2	Cluster 3	Cluster 4.1	Cluster 4.2
Obj1	neg_25.INT	$-\infty$	3	3	0	0
Obj2	neg_612.INT	$-\infty$	0	0	2	2
Obj3	cyc_25.RMS	0	0	0	1	0
Obj4	cyc_612.RMS	0	0	1	0	0
Obj5	F_model	0	0	5	4	4
Obj6	Peak_Flare_Speed	0	4	4	3	3
Obj7	sf_cyc_25.RMS_abs	0	1	0	0	0
Obj8	sf_cyc_612.RMS_abs	0	0	0	0	1
Obj9	sf_cyc_25.INT_abs	0	0	0	0	0
Obj10	sf_cyc_612.INT_abs	0	2	2	0	0
Con1	model_bdry	0	5	6	5	5
Con2	cluster_bdry	0	5	6	5	5
Con3	cluster_bdry	0	n/a	n/a	5	5

Stage 3

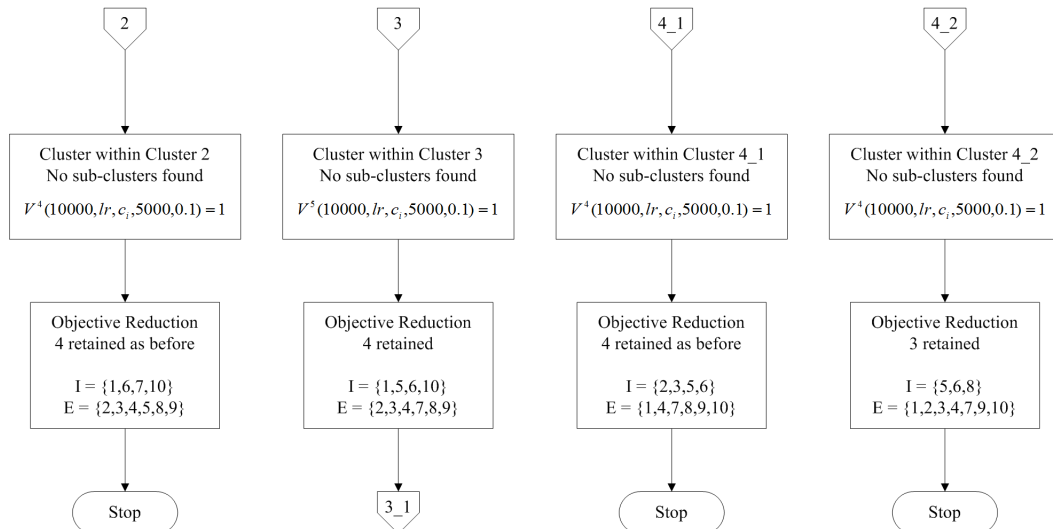


Figure 6.8: Flowchart of the results from objective reduction - 3rd stage, leading to Clusters 2, 3, 4.1 and 4.2 with 4, 4, 4 and 3 objectives being retained respectively. The objective reduction process has been concluded in all clusters, except Cluster 3. This flowchart starts after the optimisation in the 3rd stage has been completed.



and covariance matrices), which should require less learning. After examination of the results, no evidence of any converged clusters was found. Therefore:

$$\text{For Cluster 2: } V^4(10000, lr, cs, 5000, 0.1) = 1 \quad (6.12)$$

$$\text{For Cluster 3: } V^5(10000, lr, cs, 5000, 0.1) = 1 \quad (6.13)$$

$$\text{For Cluster 4.1: } V^4(10000, lr, cs, 5000, 0.1) = 1 \quad (6.14)$$

$$\text{For Cluster 4.2: } V^4(10000, lr, cs, 5000, 0.1) = 1 \quad (6.15)$$

### 6.8.3 PCA and Potential Objective Reduction - 3rd Stage

For the final populations from further optimisation with reduced objectives - 3rd Stage, FAST-MCD was used to determine robust correlation matrices, on which to base PCA and as before, the proportion of the population retained was 95%. Subsequently, PCA was conducted and the results are shown in Table H.1 in Section H.1. The grey-shaded eigenvector coefficients indicate the retained objectives resulting from application of the objective reduction rules.

A similar process to that in Section 6.6.2 was used to explore what effect varying the cumulative percentage of total variation had on the number of objectives retained. In order to achieve this the Objective reduction Rules OR3a)-3c) were applied, the results of which are shown in Table 6.8. Objective reduction was only achieved in Cluster 4.2 with a 95% threshold for the cumulative percentage of variation. Nevertheless, an 85% threshold was also evaluated to explore any further objective

Table 6.8: 3rd Stage retained objectives from objective reduction process for varying thresholds of cumulative percentage of total variation. The objective reduction selected per cluster is shown highlighted in grey. The number of retained objectives from the first stage is shown in brackets for each cluster.

Threshold for cumulative % of total variation	No. of objectives retained			
	Cluster 2 (4)	Cluster 3 (5)	Cluster 4.1 (4)	Cluster 4.2 (4)
99	4	4	4	4
95	4	4	4	3
85	4	4	4	3

reduction opportunity. The application of the Objective reduction Rules using the notation is described in Section H.1.

As described in Section 6.6.1 for the Clustering verification Rules, the use of mathematical notation for the clustering and PCA-based objective reduction process is not only compact, but can be coded in software. This will make the process more efficient, but also prevent errors, such as in the process of determining objectives to retain or discard. An example of such an error is detailed below in the process of applying the objective reduction rules within Cluster 3 for an 85% threshold for the cumulative percentage of variation. This error was not initially identified, as the analysis was carried out and completed before the results were written up. It was not possible to re-do the analysis with the error corrected as the clustering and pMOEA software used was running on a trial software license, which had expired by the time the error was found. The software implementation of the mathematical notation is the subject of future work.

#### Objective Reduction Rules applied within Cluster 3.

1. Applying OR1,  $\lambda_1/\lambda_{10} = 947.9$ , so there was not sufficient evidence of a near-linear dependency.
2. Applying OR2,  $t_2 = 85\%$ , *i.e.* the first two PCs were retained, which accounted for approximately 85% of the cumulative percentage total variation.
3. Applying OR3a),  $\theta_5 = 0.4472$ .
4. Applying OR3c) to PC1,  ${}_1A_-(1) \wedge {}_1A_+(6) \Rightarrow \{1, 6\} \subseteq I$ .
5. Applying OR3b) to PC2,  ${}_2A_-(5) \Rightarrow \{5\} \subseteq I$ .
6. For PC2, the sign of the eigenvector coefficient for Objective 10 was mistaken as being positive (it is in fact, negative) and so OR3c) was erroneously applied to PC2,  ${}_2A_-(5) \wedge {}_2A_+(10) \Rightarrow \{5, 10\} \subseteq I$ . (The correct rule to apply to PC2 was OR3b),  ${}_2A_-(5) \Rightarrow \{5\} \subseteq I$  and so, only three objectives should have been retained).

In summary (albeit in error), four objectives were retained in Cluster 3:

- neg\_25\_INT

- F\_model
- Peak\_Flare\_Speed
- sf\_cyc\_612\_INT\_abs

It should be noted that for PCs retained to account for a threshold of 99% of the total variation, in Cluster 4\_1 PC2, the magnitude of the eigenvector coefficient for Objective 5 is 1.0000 and near-zero for the other objectives and the same objective in PC1 and PC3. This suggests Objective 5 is independent, can be removed from any further multi-objective optimisation in Cluster 4\_1 and can be formulated as a separate single objective optimisation. Strong evidence of this independence can also be seen in PC3 in the same cluster as well as all of the other clusters with the 99% threshold. Such independence will spawn more optimisation problems, but of lower dimension (in this case only 2 or 1), for which it is much easier to search for and identify an optimal solution. However, when the threshold is reduced to 85% for Cluster 3 and 95% for the other clusters, evidence of this single objective independence disappears. This is because the PCA transformation projects the data onto fewer Principal Components (or Principal Axes) and the variation in the data that previously dominated a Principal Component (as indicated by the magnitude of the eigenvector coefficient) is now shared between coefficients. For example, for a 95% threshold in PC2, the variance is shared by the eigenvector coefficients for Objectives 5 and 6, whereas for a threshold of 99% the same PC2 was dominated by Objective 5. In summary, this stage culminated in:

- Two Clusters (2 and 4\_1), where there was no further objective reduction.
- Cluster 4\_2, where, although further objective reduction was achieved and subsequent optimisation was conducted, the number of objectives had reduced such that visualisation of the resulting POP was straightforward. Consequently, the objective reduction process in this cluster was concluded.
- Cluster 3, where further objective reduction was achieved resulting in a 4th stage of the objective reduction process within this cluster.

In addition, as evidenced by the independence between objectives observed in this 3rd Stage, it should be noted that reducing the threshold for the cumulative percentage of

total variation and hence the number of PCs, does not always result in the simplest possible problem structure for subsequent optimisations. For example, as in Cluster 3, one two-objective optimisation and two single-objective optimisations are considered simpler in structure than one four-objective optimisation. Nevertheless, as in Section 5.3.2, it was decided in the interests of consistency to leave this independence-based simplification for future work and persevere with objective reduction based on exploiting objective harmony.

Table 6.9: Retained objectives from objective reduction process with associated goals and priorities consistent with those specified previously in Table 6.3, where Obj1, Con1 and n/a refer to Objective 1, constraint 1 and not applicable respectively.

Objective/Constraint				
No.	Name	Goals	Priorities	
			Cluster 3_1	Cluster 4_2_1
Obj1	neg_25_INT	$-\infty$	2	0
Obj2	neg_612_INT	$-\infty$	0	0
Obj3	cyc_25_RMS	0	0	0
Obj4	cyc_612_RMS	0	0	0
Obj5	F_model	0	4	3
Obj6	Peak_Flare_Speed	0	3	2
Obj7	sf_cyc_25_RMS_abs	0	0	0
Obj8	sf_cyc_612_RMS_abs	0	0	1
Obj9	sf_cyc_25_INT_abs	0	0	0
Obj10	sf_cyc_612_INT_abs	0	1	0
Con1	model_bdry	0	5	4
Con2	cluster_bdry	0	5	4
Con3	cluster_bdry	0	n/a	4

## 6.9 Results from the Objection Reduction Process - 4th Stage

### 6.9.1 Further Optimisation with Reduced Objectives - 4th Stage

Two subsequent optimisations were carried out with the reduced objectives from Stage 3. In Cluster 3 the number of objectives was reduced from five to four, now denoted as Cluster 3\_1. In Cluster 4\_2 an objective reduction of four to three was achieved, denoted as Cluster 4.2\_1. In the latter case, it is straightforward to visualise a three-dimensional Pareto-optimal front and select a preferred solution and so further objective reduction was not pursued.

Previous optimisation parameter settings were used as defined in Section 6.7.1. The objective goals and priority order have been inherited from the previous objective reduction process (see Table 6.7) and are shown in Table 6.9.

A summary of the 4th stage objective reduction process after the optimisation is provided in the flowchart in Figure 6.9.

### 6.9.2 Clustering and Verification with Reduced Objectives - 4th Stage

In Cluster 3\_1 only, the final population resulting from the further optimisation with reduced objectives - 4th Stage, was clustered using the k\*-Means clustering algorithm. The initial parameters were set as previously in Section 6.6.1, except that now the number of runs was set to 35 (corresponding to the number of processors available to the pMOEA). No evidence of any converged clusters was found, *i.e.*

$$V^4(10000, lr, cs, 5000, 0.1) = 1 \quad (6.16)$$

### 6.9.3 PCA and Potential Objective Reduction - 4th Stage

For the final populations resulting from the further optimisation with reduced objectives - 4th Stage, FAST-MCD was used to determine robust correlation matrices, on which to base PCA and as before, the proportion of the population retained was 95%. Subsequently, PCA was conducted and the results are shown in Table I.1 in Section I.1. The grey-shaded eigenvector coefficients indicate the retained

objectives resulting from application of the objective reduction rules.

A similar process to that in Section 6.6.2 was used to explore what effect varying the cumulative percentage of total variation had on the number of objectives retained. In order to achieve this, the Objective reduction Rules OR3a)-3c) were applied for 95% and 85% thresholds for the cumulative percentage of variation. The application of the Objective reduction Rules using the notation is described in Section I.1.1. No further objective reduction was achieved with either threshold.

While the same mistake was made again as in the 3rd stage in Cluster 3 (see Section 6.8.3), it is reassuring that after the 4th stage optimisation and objective reduction the same pattern (objectives retained and eigenvector coefficient magnitudes) has resulted in Cluster 3 with the lower threshold for cumulative percentage of total

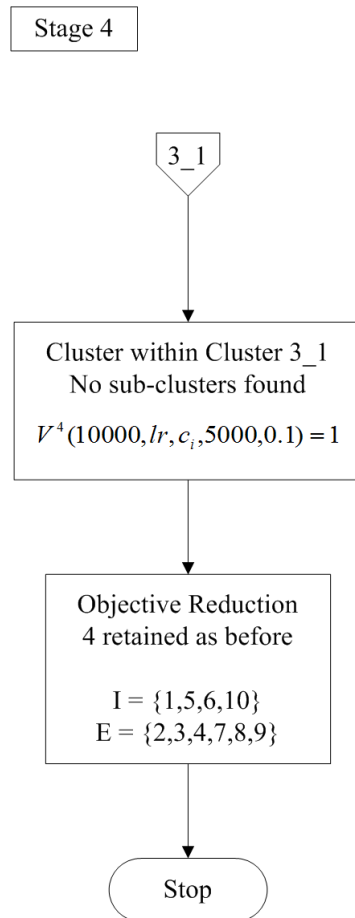


Figure 6.9: Flowchart of the results from objective reduction - 4th stage in Cluster 3\_1, resulted in no further objective reduction. This flowchart starts after the optimisation in the 4th stage has been completed.

variation. (Note: whilst a 95% was used as the lower threshold in stage 3, compared to an 85% threshold in Stage 4, had the same 85% been used in Stage 3, the results would not have changed as it would still be two PCs that would have been retained).

## 6.10 Conclusions from the Objective Reduction Process

The results from the final objective reduction from the 3rd Stage are displayed in Tables 6.10 and 6.11, where the latter shows which objectives were retained in each cluster at each stage. The results from the 3rd Stage show that, in general, the high priority objectives were retained, which is not surprising given that objective priority is accounted for in the objective reduction rules.

Parallel coordinates plots of the POPs per cluster resulting from the final objective reductions are shown in Figure 6.10 and 6.11. These plots were reviewed with the Cold Start calibration engineer and the following conclusions were arrived at:

- In Cluster 3 (green), the F Objective was in the range 1.5-1.6. At these low levels of Fuel quantity, the engine cold start performance was erratic when lower quality fuels (available in markets such as Russia) were tested.
- In Cluster 3, while the n612I Objective was relatively high, (combustion intensity for cycles 6-12 was strong), this was at the expense of the n25I Objective, which was comparatively low (weak combustion intensity for cycles 2-5).
- In Figure 6.11, it can be seen that in Cluster 3 when the sensitivity Objective

Table 6.10: Number of objectives retained at each stage of the objective reduction process.

Objective reduction Stage	No. of objectives retained			
	Cluster 2	Cluster 3	Cluster 4	
1st	6	7	5	
			Cluster 4_1	Cluster 4_2
2nd	4	5	4	4
3rd	4	4	4	3

Table 6.11: Table of which specific objectives and the number of them retained at each stage of the objective reduction process per cluster, where retained objectives are shaded grey and objective names are further abbreviated for plotting purposes. The priority order is that used by the Comparison Operator in the modified NSGAII such that a high number is a high priority and *vice versa*.

Objective				Objective Reduction Stage											
No.	Name	Abbreviation	Priority	1st			2nd				3rd				
				Cluster 2	Cluster 3	Cluster 4	Cluster 2	Cluster 3	Cluster 4.1	Cluster 4.2	Cluster 2	Cluster 3	Cluster 4.1	Cluster 4.2	
Obj1	neg_25_INT	n25I	4												
Obj2	neg_612_INT	n612I	4												
Obj3	cyc_25_RMS	25R	2												
Obj4	cyc_612_RMS	612R	2												
Obj5	F_model	F	6												
Obj6	Peak_Flare_Speed	PFS	5												
Obj7	sf_cyc_25_RMS_abs	s25R	1												
Obj8	sf_cyc_612_RMS_abs	s612R	1												
Obj9	sf_cyc_25_INT_abs	s25I	3												
Obj10	sf_cyc_612_INT_abs	s612I	3												
Number of Objectives Retained				6	7	5	4	5	4	4	4	4	4	3	

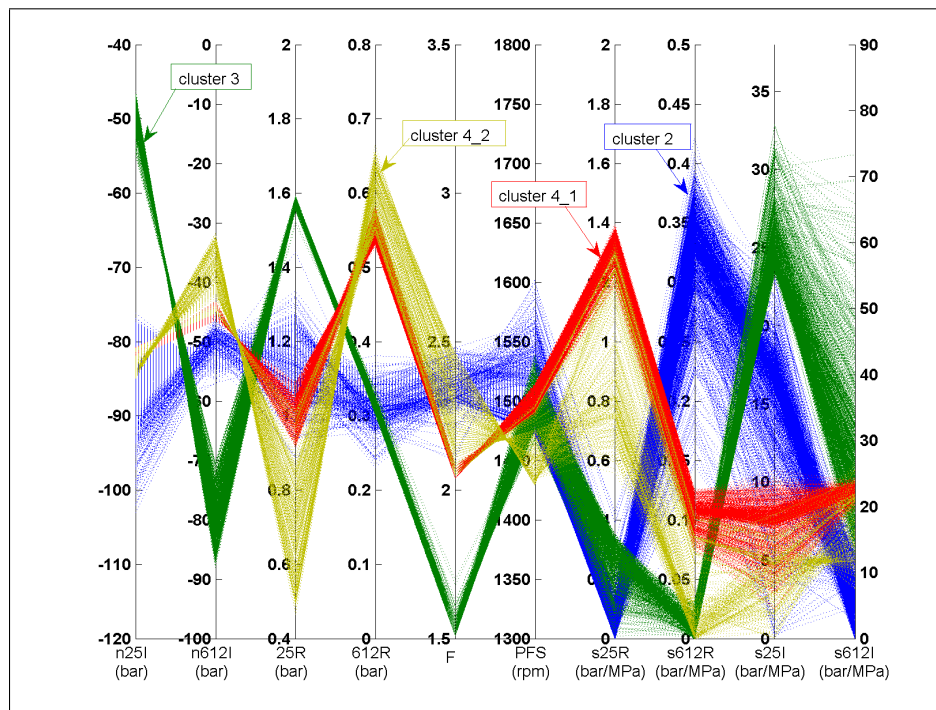


Figure 6.10: Parallel Coordinates plot of the final POPs resulting from objective reduction in each cluster. See Table 6.11 for objective abbreviations.



s612I is plotted next to the Objective to which it relates, n612I, the resulting sensitivity values are in some cases almost as large for those for n612I, indicating that these solutions show high sensitivity.

- The resulting POPs in Clusters 4.1 and 4.2 display somewhat similar performance, which is to be expected given that they have the same parent cluster. Nevertheless, it can be seen that Cluster 4.1 (red) performs worse in the sensitivity objectives than Cluster 4.2 (gold). In this case, more sensitivity means the start performance is less robust to variations in Fuel Pressure, which is not tightly controlled. In a mass-production environment, this variation is likely to increase and may lead to poor customer satisfaction with start performance and potentially, warranty cost.

Consequently, it was decided in consultation with the calibration engineer to discard Clusters 3 (green) and 4.1 (red) and to select preferred solutions from the retained Clusters 2 (blue) and 4.2 (gold).

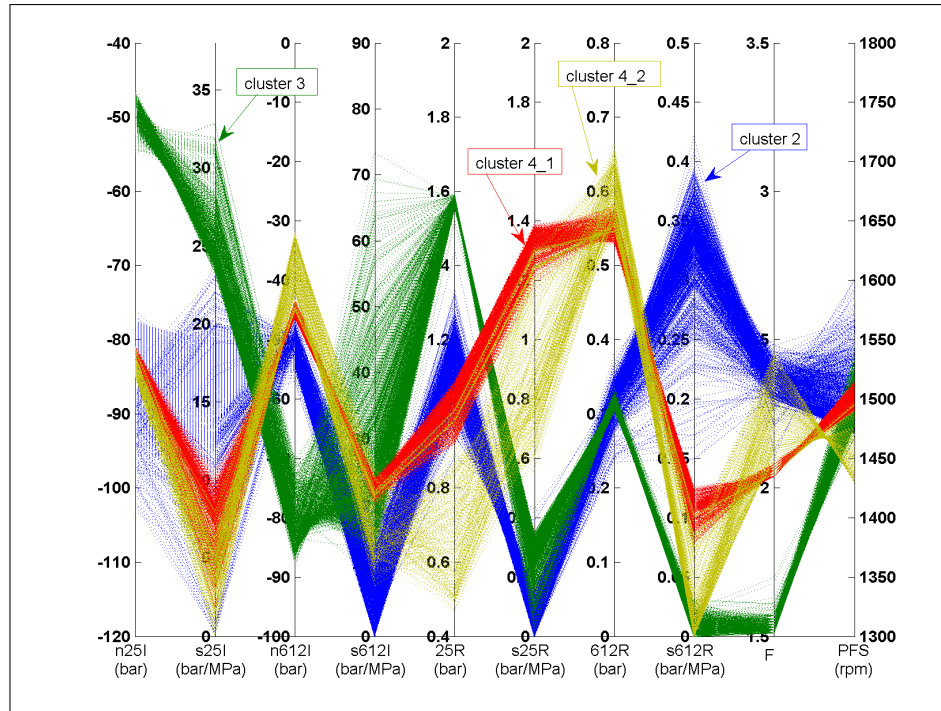


Figure 6.11: Parallel Coordinates plot of the final POPs resulting from objective reduction in each cluster. The objectives have been re-ordered so that each sensitivity objective is adjacent to the objective to which it relates.

### 6.10.1 Selection of Final Solutions

In Cluster 2, of the retained Objectives, PFS (Peak\_Flare\_Speed) was the highest priority. Using this information and a brushing technique, available in generating figure plots in Matlab® v7.7 (R2008b), a preferred solution that was relatively insensitive as measured by s25R and s612I and also with a relatively low value of F (Fuel quantity - a surrogate measure for HC emissions) was selected. This solution is shown in red in objective space in Figure 6.12 and in decision variable space in Figure 6.13. As can be seen in the plot of PFS versus n25I (top left) in Figure 6.12, there was evidence of a strong trade-off or conflict between these two objectives. This complies with known physical understanding of engine starting, as a stronger NMEP in cycles 2-5 (more negative n25I) indicates that the combustion intensity is stronger just after the engine has fired, which will in turn cause the engine speed to rise to a higher level (or higher PFS). The converse is also true.

In Cluster 4.2, of the retained objectives, the highest priority objective was F (Fuel quantity), then PFS, then the s612R (sensitivity of cycles 6-12 combustion variation to fuel pressure). Using this information and brushing, a preferred solution that is

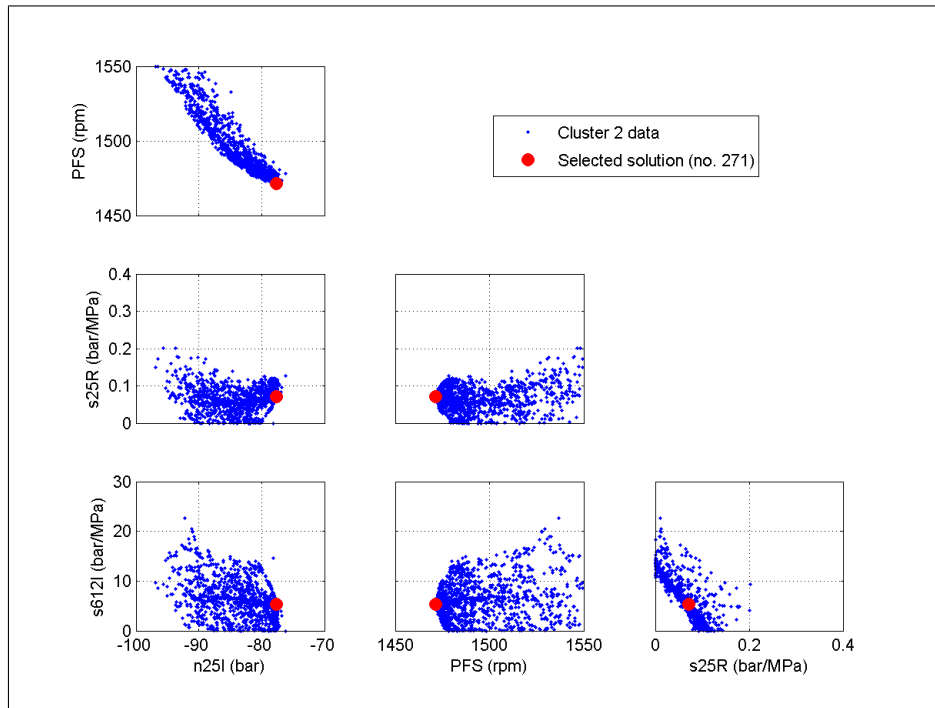


Figure 6.12: Scatter plot of the objectives in the final POP (blue) resulting from objective reduction in Cluster 2 and the selected solution (red).

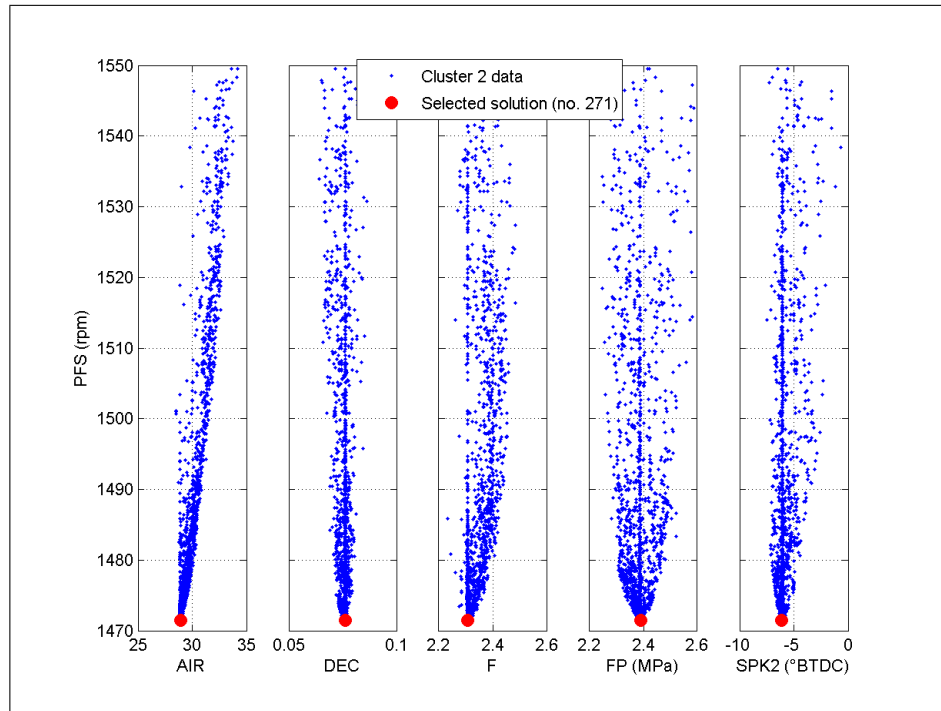


Figure 6.13: Scatter plot of the decision variables in the final POP (blue) resulting from objective reduction in Cluster 2 and the same selected solution (red) as in Figure 6.12.

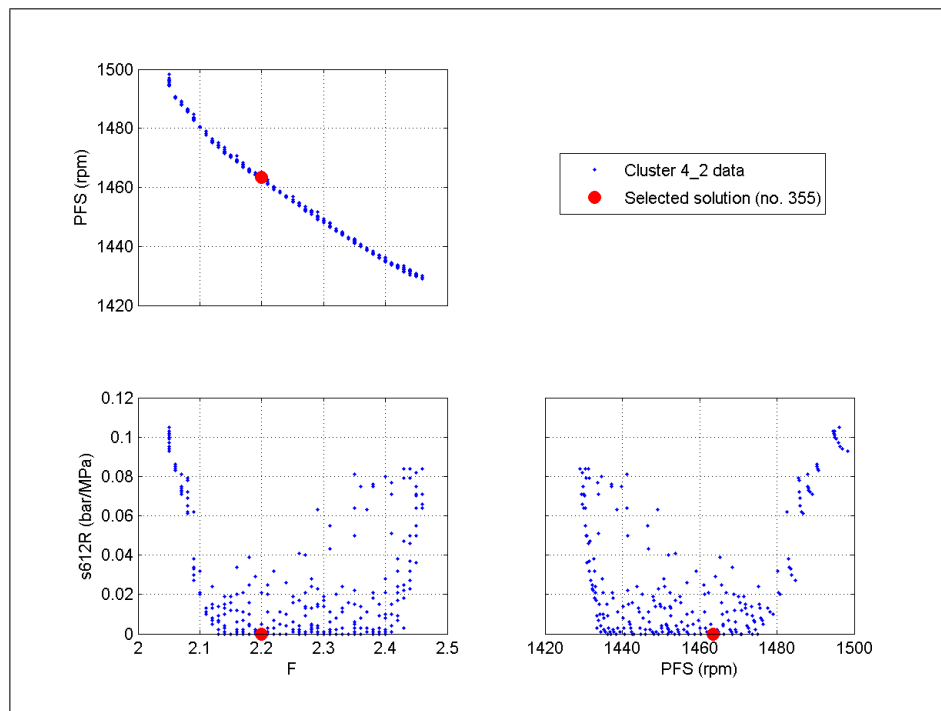


Figure 6.14: Scatter plot of the objectives in the final POP (blue) resulting from objective reduction in Cluster 4.2 and the selected solution (red).

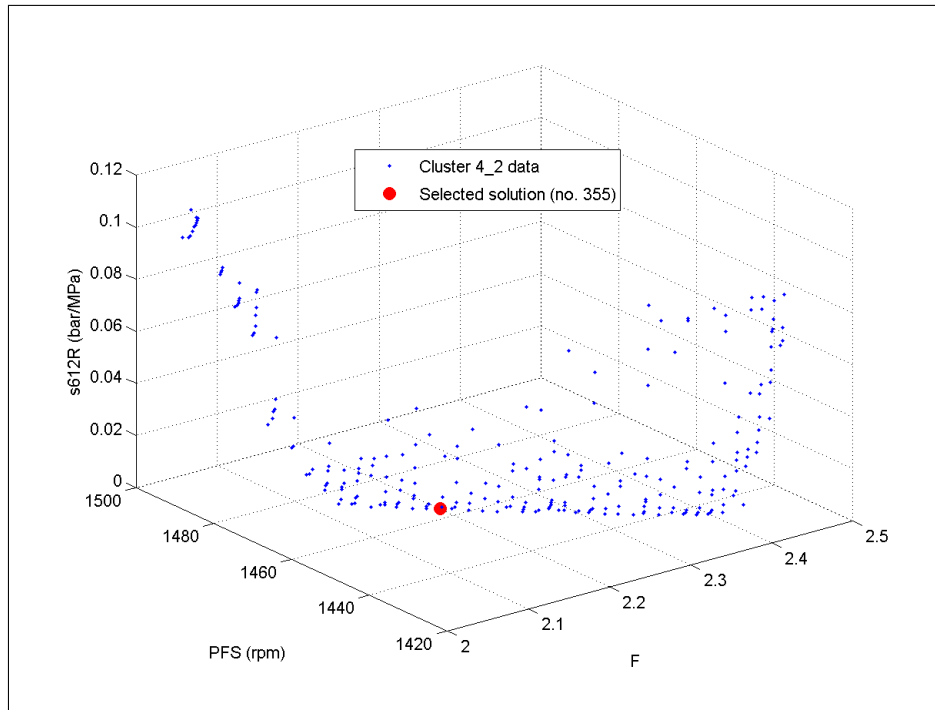


Figure 6.15: Scatter plot in 3D of the objectives in the final POP (blue) resulting from objective reduction in Cluster 4.2 and the selected solution (red).

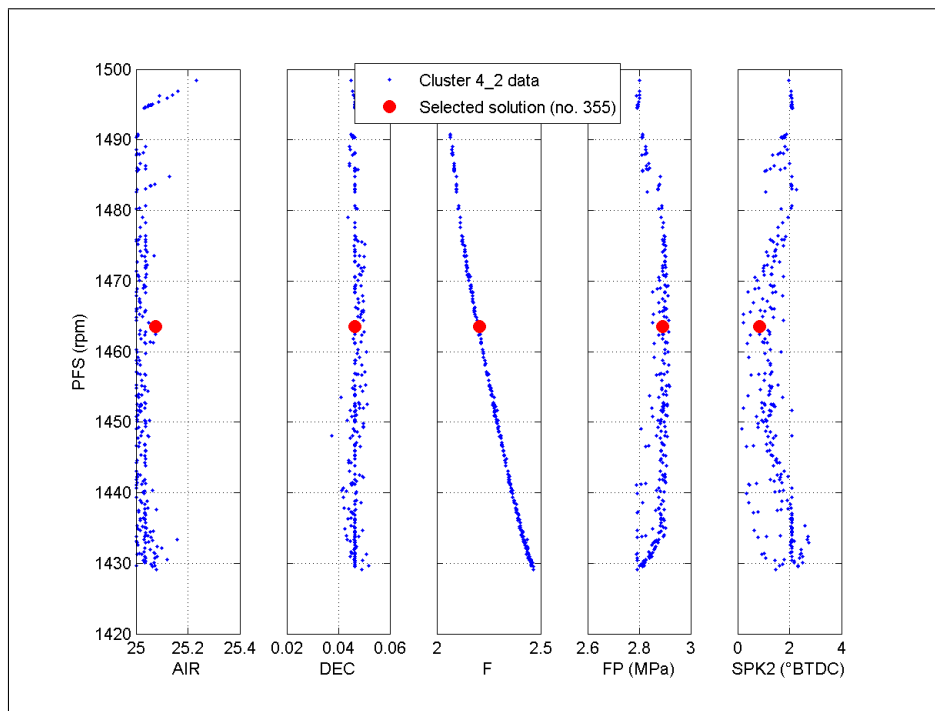


Figure 6.16: Scatter plot of the decision variables in the final POP (blue) resulting from objective reduction in Cluster 4.2 and the same selected solution (red) as in Figure 6.14.

relatively insensitive as measured by s612R was selected. This solution is shown in red in objective space in Figures 6.14 and 6.15 and in decision variable space in Figure 6.16. The chosen solution was identified principally from the scatter plot of F versus PFS (top left) in Figure 6.14, where F was favoured slightly over PFS, but also had a low s612R. This can be seen more clearly in the three-dimensional scatter plot in Figure 6.15.

The objective values of the final solutions selected from each cluster are shown in Table 6.12 with the retained objectives in each cluster highlighted in grey and the remaining objectives discarded. Comparing the two solutions, initial observations were:

- In the objectives where at least one was retained (shaded grey), the retained objectives generally performed better than the discarded ones. The exception being the neg\_25\_INT Objective where solution 355 was better (larger negative value), but only by a relatively small extent. In the Peak\_Flare\_Speed Objective both solutions were similar in value. This improved performance in the retained objectives was expected as these were the objectives retained for further optimisation.
- In the other objectives, which have been discarded for both solutions, solution 271 was better than 355 in neg\_612\_INT and cyc\_612\_RMS, but worse in cyc\_25\_RMS and sf\_cyc\_25\_INT\_abs.
- In other words, each solution was neither better nor worse than the other, just different.

Of further interest was a comparison of these solutions against their respective parent cluster of solutions and against a recent calibration generated by the Cold Start

Table 6.12: Final solutions selected from retained Clusters 2 and 4.2. The objectives retained are shown highlighted in grey.

Cluster	Sol. no.	n25I	n612I	25R	612R	F	PFS	s25R	s612R	s25I	s612I
2	271	-77.69	-54.38	1.225	0.317	2.31	1471.5	0.071	0.326	16.73	5.4
4.2	355	-83.12	-39.41	0.769	0.59	2.2	1463.5	0.812	0	3.21	14.12

calibration engineer using a manual, iterative tuning process.

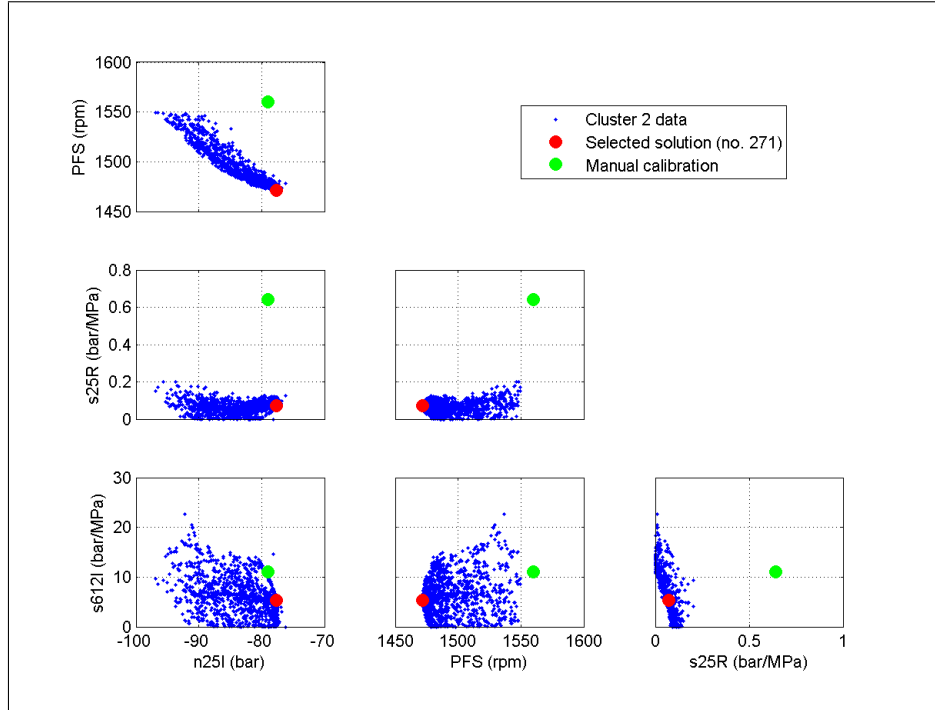


Figure 6.17: Scatter plot of the retained objectives in the final POP (blue) resulting from objective reduction in Cluster 2, the selected solution, no. 271 (red) and the independently developed, manually tuned calibration (green).

Scatter plots of this data are shown for Cluster 2 and 4.2 in Figure 6.17 and Figure 6.18, respectively. In addition, parallel coordinates plots of this data are shown for Cluster 2 and 4.2 in Figure 6.19 and Figure 6.20, respectively. For each figure, the final POP (blue) resulting from objective reduction, the selected solution (red) and an independently, manually generated calibration (green) are overlaid. For both clusters, it can be seen that the calibration is inferior with respect to the POP and the selected solutions. The exception is in Cluster 2, where the calibration is slightly better (smaller) than selected solution 271 with respect to the neg\_25\_INT Objective. In summary, the selected solution is a significant improvement compared to the calibration in respect of:

- **In Cluster 2.** Peak\_Flare\_Speed (PFS) and combustion variation sensitivity (for cycles 2-5) (s25R). The former is considered to be very important for customer satisfaction with the starting process and the latter indicates much

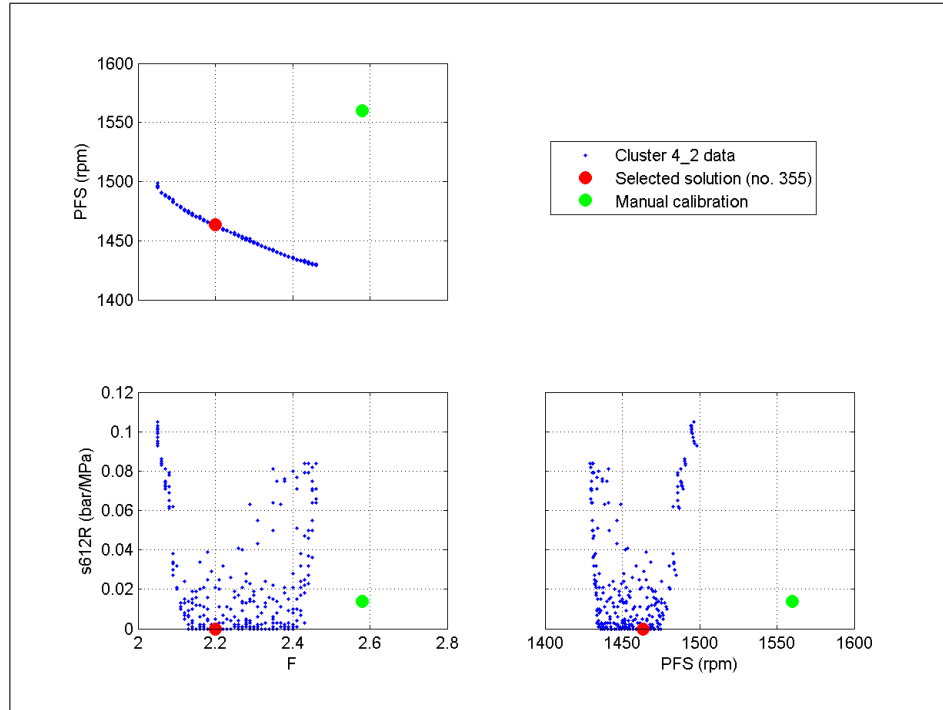


Figure 6.18: Scatter plot of the retained objectives in the final POP (blue) resulting from objective reduction in Cluster 4.2, the selected solution, no. 355 (red) and the independently developed, manually tuned calibration (green).

improved start robustness and quality.

- **In Cluster 4.2.** Fuel quantity (F) and PFS. Fuel quantity is very important with regard to ever-increasing customer expectations of good fuel economy. Also, as fuel quantity has been used as a surrogate for legislated HC emissions, reducing fuel reduces HC. So, in this case, significantly reduced HC emissions is considered especially important as the vast majority of HC emissions are produced before the exhaust aftertreatment system (*e.g.* catalytic convertor) has reached operating temperature, *i.e.* at and after engine start.

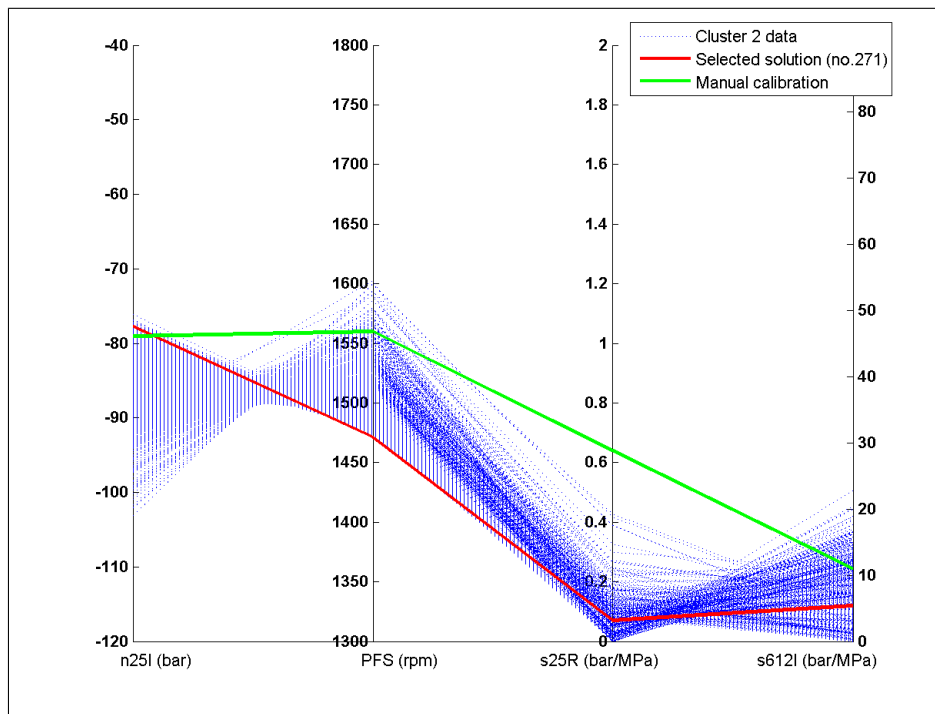


Figure 6.19: Parallel coordinates plot of the retained objectives in the final POP (blue) resulting from objective reduction in Cluster 2, the selected solution, no. 271 (red) and the independently developed, manually tuned calibration (green).

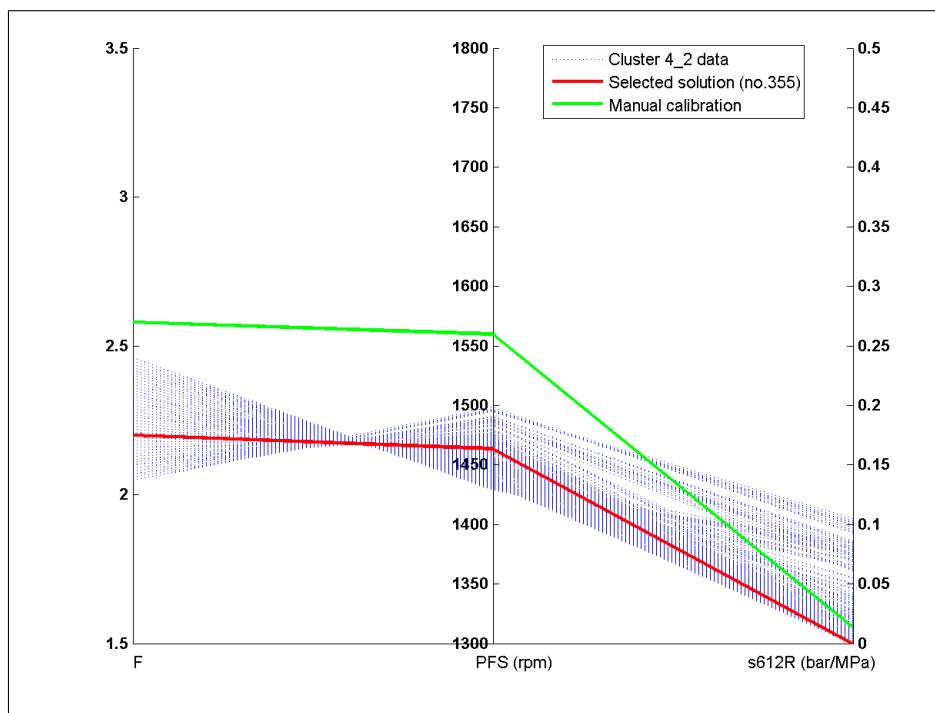


Figure 6.20: Parallel coordinates plot of the retained objectives in the final POP (blue) resulting from objective reduction in Cluster 4.2, the selected solution, no. 355 (red) and the independently developed, manually tuned calibration (green).



## 6.11 Summary

In this case study a real-world transient calibration problem has been distilled into a ten-objective many-objective optimisation problem. The objectives include dynamic measures of combustion quality as well as sensitivity quantities related to a control system actuator, which exhibits significant variation. To address the computational demands of such a high-dimensional problem, use was made of a parallel computing cluster. The modified NSGAI multi-objective genetic algorithm of Chapter 4 was parallelised using an island-based approach. In addition, the k\*-Means clustering algorithm was run in batch mode allowing many more clustering runs to be carried out. A mathematical notation for the Clustering verification and Objective reduction Rules was introduced for brevity and efficiency and to enable future software development. The objective reduction process consisted of four stages and progressively reduced objective dimensionality where evidence of local objective harmony existed. It involved the calibration engineer at various stages to advise on objective priorities and to discard clusters containing solutions of no interest. This process culminated in two sub-problems, one of three and one of four conflicting objectives. From the corresponding POPs, scatter plots and brushing, together with priorities for the remaining objectives, were used to identify preferred solutions. A comparison of the resulting POP, preferred solution and an independently generated, manually tuned calibration was made for each of the two sub-problems. In general, the preferred solution outperformed the independent calibration.



## Chapter 7

# Conclusions and Future Work

### 7.1 Conclusions

The objectives of this research were to apply multi-objective evolutionary algorithms to many-objective real-world problems and in doing so, to develop a decision-making process to ultimately select a single preferred solution utilising an intuitive results presentation.

To this end, a novel process has been developed for reducing complexity in high-dimensional, real-world, multi-objective optimisation problems. This approach relies on the principle of being able to identify and exploit local harmony between objectives to reduce dimensionality. To achieve this, a systematic and modular process has been designed to partition the Pareto-optimal front into clusters, within which a rule-based Principal Component Analysis (PCA) including preference articulation is applied for potential objective reduction. This process has been applied to several real-world automotive engine problems of increasing complexity from three to ten objectives. In each case study, the number of objectives was progressively and systematically reduced and a preferred, better-performing solution selected.

Conclusions from this thesis are described below and correspond to the numbered list of Contributions defined in Chapter 1.

1. **A novel, systematic and modular dimension reduction process.** This is a sequential process of applying search, clustering and PCA-based objective reduction elements, as depicted in the flowchart in Figure 4.1. Rules have been developed to verify the clustering analysis and identify any objective reduction

opportunity as detailed in Sections 4.4.3 and 4.5, respectively. No claim is made that each of the methods used for elements are the best for their respective tasks, but those used in this research programme have provided useful results on real-world problems, as demonstrated in the simplified example in Section 4.7 and the case studies in Chapters 5 and 6. The application to these examples of Pareto-front partitioning and dimensionality reduction coupled with visualisation enabled the decision-maker to select solutions, which outperformed those chosen using a manual tuning approach. Nevertheless, each one of the elements can be replaced by an alternative, which may have advantages in a given application such as being superior in performance, more readily available or more familiar to the researcher.

## 2. Novel exploitation of local objective harmony for dimension

**reduction.** The aforementioned dimension reduction process relies on the principle of being able to identify and exploit any local harmony between objectives for potential dimension reduction. The principle is based on the observations detailed in Section 2.6 that:

- Objective harmony and conflict may vary across the Pareto-optimal front (Purshouse and Fleming, 2003a; Deb and Saxena, 2005). The clustering element is used to group like-solutions in the population to allow any local objective dependencies to be explored.
- If a subset of the objectives in the Decision Maker's region of interest are in harmony, a local objective reduction opportunity exists (Purshouse, 2003; Deb and Saxena, 2006). The PCA-based analysis is used to identify objective dependencies for potential objective reduction.

There are three further contributions related to using local objective harmony for dimension reduction as follows:

- (a) **Definition of clustering verification rules.** The  $k^*$ -Means clustering algorithm used in this research programme is stochastic by nature, which justifies running it several times to give confidence that the results are reliable. In addition, it is necessary that the Decision Maker (DM) is satisfied that the number and location of the resulting clusters is consistent

and correct. Consequently, a number of rules (see Section 4.4.3) have been devised to carry out the clustering with varying initial settings as well as a sub-sampling approach to identify the smallest, most computationally efficient population suitable for clustering. Furthermore, the resulting cluster membership and bounds are compared to justify retaining all clusters. The latter was employed with notable effect in the Cold Start optimisation: 1st Stage Objective Reduction (Section 6.6.1), where data within one cluster violated an engineering limit and was subsequently discarded.

- (b) **Definition of objective reduction rules.** In problems comprising many objectives, it may be possible to find a subset containing the majority of information. Rules to determine this subset have been proposed (in Section 4.5) to objectively and systematically identify and retain the most influential objectives within each retained Principal Component.

Both of these sets of rules have been applied in the simplified example in Section 4.7 and the case studies in Chapters 5 and 6 and have led to significant objective reduction.

- (c) **A new mathematical notation for the clustering and objective reduction rules.** With problems comprising a larger number of objectives, it is possible that the number of stages of the objective reduction will increase and application of the rules will become lengthy. This was borne out by the four stages involved in the Cold Start optimisation in Chapter 6. To address the lengthy application of the rules, mathematical notations were developed to provide clarity, brevity and efficiency. These notations are defined in Appendices D and E for the clustering verification and objective reduction rules, respectively. In addition, they are applied in Chapter 6, the results from which are summarised in flowcharts for each stage of the objective reduction process, as shown in Figure 6.6, for example.

3. **Inclusion of sensitivity objectives in the optimisation.** This allows simultaneous and efficient search for solutions providing optimal trade-offs between maximising performance and minimising sensitivity to background

noise, as exists in the Cold Start optimisation in Chapter 6. The Direct Derivatives method most closely matches the requirements of being a computationally efficient, easy-to-implement sensitivity analysis approach, which can be integrated into the optimisation and is sufficiently general for engine calibration optimisation problems. The empirical model-based approach used in this thesis made it straightforward to develop such a Direct derivative approximation without the need for additional test data. Four such sensitivity objectives were developed and included in the Cold Start optimisation.

Further minor contributions that resulted from the research work in this thesis are:

- i. **Objective priority use in objective reduction.** Objective priorities and goals were specified in both of the six and ten objective problems in Chapters 5 and 6, respectively. These were not only implemented with the modified NSGAI to guide the search, but also were directly applied to the PCA-based objective reduction process in both case studies to discriminate between objectives with a similar influence. In the Cold Start optimisation (Chapter 6), objective priorities were provided by the Cold Start calibration engineer.
- ii. **Flexibility in reducing the number of objectives.** A study on the effect of varying the threshold for selecting PCs (see Section 5.5) showed that this had a significant effect on the number of objectives retained. In the Cold Start optimisation problem in Chapter 6, this effect was utilised to achieve more objective reduction for a slightly lower threshold for retaining PCs and as such, may provide some objective reduction flexibility to the DM.
- iii. **Inclusion of pertinency in a MOEA.** The widely-used Multi-Objective Evolutionary Algorithm, NSGAI (Deb, Pratap, Agarwal and Meyarivan, 2002) has been modified to incorporate the progressive preference articulation method of Fonseca and Fleming (1998a). To achieve this, it was necessary to gain a good understanding of how this method works, as documented in Appendix B. The modified NSGAI not only allows the decision maker to zoom in to the region of interest, but also shrinks the search space in high dimensional problems. This is achieved by intuitively specifying objective priorities and goals and has been used in all the real-world problems analysed in this research programme.

- iv. **New parallel MOEA.** A new island-based parallel version of the modified NSGAII has been developed for efficient evaluation of large populations. This was implemented as it is one approach for overcoming the search effectiveness issues of the serial NSGAII and a compute cluster comprising 35-40 processors was available. In Section 6.4, a validation test conducted to verify that the implemented pMOEA generated broadly equivalent Pareto-Optimal Populations (POPs), showed that the serial and parallel MOEAs corresponded well in terms of objective pattern and distribution. The pMOEA was applied to the multi-stage Cold Start optimisation in Chapter 6 and proved to be computationally efficient, *e.g.* it took 2-3 hours to run the pMOEA for 25,000 generations on a population of 10,000 for the ten-objective problem.
- v. **Parallel computing applied to clustering.** The clustering process is randomly initialised and thus needs to be run several times from different values for the initial number of clusters. This can be computationally demanding for large populations used in this thesis for high-dimensional problems. A batch processing approach exploiting a distributed computing network is essential to make this task practical for high-dimensional problems. Consequently, this approach was employed for the clustering and verification elements in each of the four optimisation stages in the Cold Start optimisation in Chapter 6.
- vi. **Use of a cluster boundary constraint.** The cluster centres and covariance data resulting from the clustering process have been used to define a hyper-ellipsoidal constraint, associated with a cluster in an optimisation in an attempt to preserve objective correlations within the cluster, as defined in Section 3.5.6. This is necessary, as it is important to verify that in subsequent optimisations, any discarded objectives do not deteriorate. This was applied in all three real-world problems in this research programme.
- vii. **Definition of dynamic measures of combustion quality.** These comprised smoothed, dynamic measures of combustion intensity and variation. Respectively, these are equivalent to maximising the energy available to accelerate the engine during the ‘run-up’ and minimising the combustion variation to give a smooth engine response. These measures were used as objective functions in the Cold Start optimisation in Chapter 6 and are defined in Section 6.5.1.

## 7.2 Future Work

Although significant objective reduction and favourable optimisation solutions have been realised with the complexity reduction process as introduced in Chapter 4, there are a number of enhancements which can be made. Such improvements have been categorised into those which address observed weaknesses or make improvements to elements of the existing proposed process and those exploring other, more general many-objective optimisation research opportunities.

- **Improvements to the existing proposed dimension reduction process**

- In the Migration of Preferability (PPA<sub>FF</sub>) to NSGAII (Section 4.3.3), the Archive population block (see Figure 4.3) uses a constrained fast non-dominated (CFND) sorted (as with NSGAII) for both the NSGAII and PPA<sub>FF</sub> paths. This use of CFND sorting is not a feature of the original PPA<sub>FF</sub> path, which instead ranks the population by preference. Therefore, the Archive population block should be extended to accommodate both paths, not just CFND. Once implemented, the relative performance of the archiving approaches should be compared to understand what effect such a change would have.
- Revisit the objective reduction analyses and subsequent optimisation in both case studies where there is evidence of independence between objectives. This is indicated by a retained Principal Component (PC) in which a particular objective dominates, as evidenced by a near-unity eigenvector coefficient magnitude with corresponding near-zero eigenvector coefficient magnitudes for the same objective in the other PCs. In such scenarios, an increased number of lower dimension optimisations may result.
- Explore and compare the performance of alternative migration parameter settings and schemes for the island-based pMOEA implementation as described in Section 6.4.
- While the k\*-Means clustering algorithm automatically determines the number of output clusters, its parameters may require a lot of calibration, *i.e.* lots of runs, to generate converged clusters. Other clustering



approaches should be tried. For example, it may be possible to comply with the spherical data assumption of the k-Means algorithm (Macqueen, 1967), by normalising objective function inputs and then running it many times (particularly if parallel computing is available) from different numbers of initial clusters. Comparison of the cluster convergence should reveal the correct number of output clusters.

- As demonstrated in the case studies in Chapters 5 and 6, the two main stages of the search (evolution of an initial population for clustering and further evolution of populations within clusters) have involved many thousands of objective function evaluations. Further research is required to determine whether this significant computational investment is justified or if fewer objective function evaluations would suffice.

- **More general many-objective optimisation research opportunities**

- Implement the mathematical notation for the Clustering verification and Objective reduction Rules in software. This will make a high dimensional multi-stage objective reduction process more efficient, less error-prone and potentially fully automated including documentation of results at each stage. Different degrees of objective reduction and different objective priority orders could easily be explored, which may be subject to available computational resources.
- This software could be implemented as a collection of routines underlying a Graphical User Interface in for example, a Matlab<sup>®</sup> toolbox. This could be designed to easily allow alternative optimisation, clustering or dimension reduction algorithms to be ‘plugged-in’ as well as being able to interface to a parallel computing facility.
- The ‘toolbox’ software concept could be extended into a more general ‘Many-Objective Optimisation’ toolbox, which not only supported the proposed dimension reduction process, but also provided for other processes, based on alternative methods to be implemented. For example, hybrid MOEAs such as the CAO-based NSGAI (Adra *et al.*, 2009) or modified Pareto-dominance algorithms such as that based on *L-dominance* (Zou *et al.*, 2008) could be included. In addition, further visualisation

capability could be incorporated, *e.g.* the Hyper-Radial Visualization tools (Chiu and Bloebaum, 2008, 2010). Furthermore, the toolbox could provide a convenient and easy-to-use capability to compare the results of various alternative approaches to optimising the same problem.

# Appendix A

## Details of k\*-Means Simulation Testing

This appendix provides details of the simulation testing on the k\*-Means clustering algorithm summarised in Section 3.3.5.

For the normal data and non-normal data a multivariate normal distribution and a  $\chi^2$  distribution were sampled, respectively. A number of scenarios have been run with the results as shown below:

Scenario:

### 1. Normal data: one cluster, two variables

This is the simplest scenario in which the clustering algorithm should work and a plot of the data sample is shown in Figure A.1. The data consisted of 200 points as a function of two variables,  $x_1$  and  $x_2$ . As detailed below, there was good agreement between the centres of the cluster,  $M_c$  and those for the data,  $M_d$ .

$$M_c = \begin{bmatrix} 0.0733 & 0.0837 \end{bmatrix} \quad M_d = \begin{bmatrix} 0 & 0 \end{bmatrix}$$

Furthermore, in lower triangular form, the correlation matrix of the cluster (determined by an iterative method in the k\*-Means of Cheung (2003)),  $C_c$ , corresponded well to that of the data,  $C_d$ , produced by the random number generator. In turn,  $C_d$  was similar to the population correlation matrix,  $C_p$  (derived from the covariance matrix specified to the multivariate, normal,

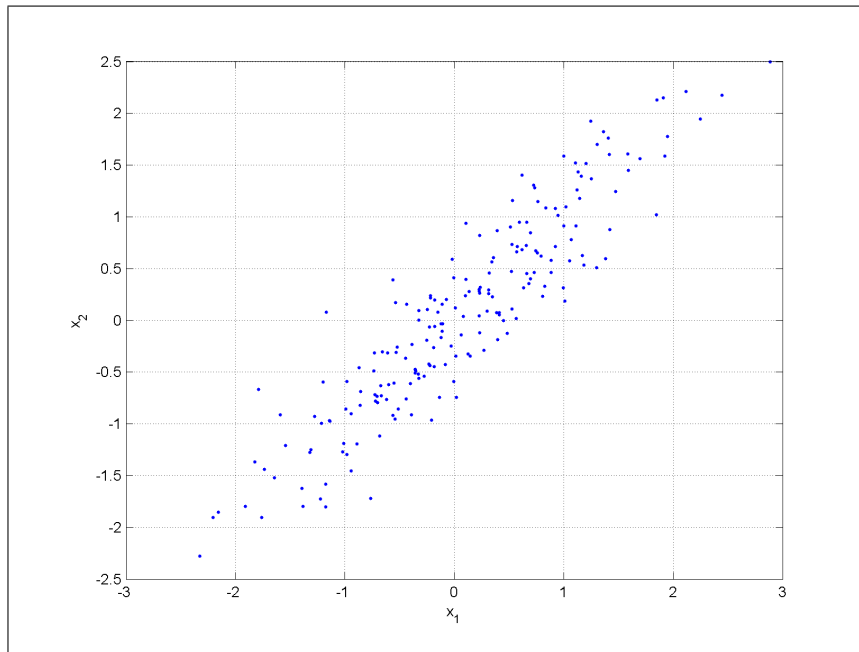


Figure A.1: Single group of normal data.

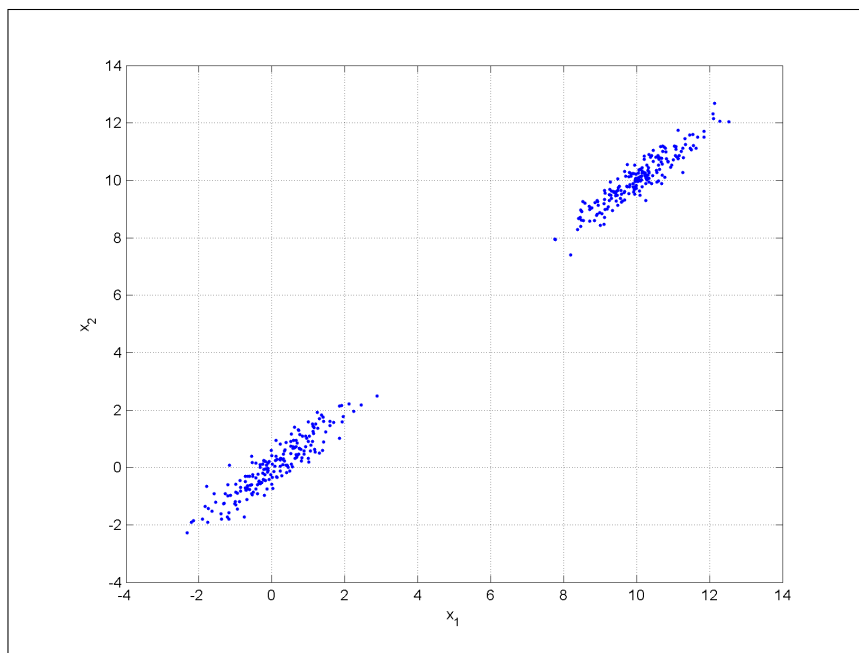


Figure A.2: Two well-separated groups of normal data.

pseudo-random number generator used to produce the data sample).

$$C_p = \begin{bmatrix} 1.0000 & \\ 0.9254 & 1.0000 \end{bmatrix} \quad C_d = \begin{bmatrix} 1.0000 & \\ 0.9196 & 1.0000 \end{bmatrix} \quad C_c = \begin{bmatrix} 1.0000 & \\ 0.9189 & 1.0000 \end{bmatrix}$$

## 2. Normal data: two well-separated clusters, two variables

The algorithm should have no difficulty in discerning between the groups of data shown in Figure A.2. Each of these groups consisted of 200 points as a function of the same two variables as before. Good agreement was achieved between the centres of each cluster,  $M_{c_{1,2}}$  and the respective centers of each data group,  $M_{d_{1,2}}$ .

$$M_{c_1} = \begin{bmatrix} 0.0099 & 0.1550 \end{bmatrix} \quad M_{d_1} = \begin{bmatrix} 0 & 0 \end{bmatrix}$$

$$M_{c_2} = \begin{bmatrix} 9.9371 & 10.2060 \end{bmatrix} \quad M_{d_2} = \begin{bmatrix} 10 & 10 \end{bmatrix}$$

The correlation matrices of each cluster,  $C_{c_{1,2}}$ , corresponded well to those of the data,  $C_{d_{1,2}}$ , produced by the random number generator. In turn,  $C_{d_{1,2}}$  were similar to the population correlation matrices,  $C_{p_{1,2}}$ .

$$C_{p_1} = \begin{bmatrix} 1.0000 & \\ 0.9254 & 1.0000 \end{bmatrix} \quad C_{d_1} = \begin{bmatrix} 1.0000 & \\ 0.9196 & 1.0000 \end{bmatrix} \quad C_{c_1} = \begin{bmatrix} 1.0000 & \\ 0.9073 & 1.0000 \end{bmatrix}$$

$$C_{p_2} = \begin{bmatrix} 1.0000 & \\ 0.9254 & 1.0000 \end{bmatrix} \quad C_{d_2} = \begin{bmatrix} 1.0000 & \\ 0.9265 & 1.0000 \end{bmatrix} \quad C_{c_2} = \begin{bmatrix} 1.0000 & \\ 0.9114 & 1.0000 \end{bmatrix}$$

## 3. Normal data: two adjacent clusters, two variables

This is a more realistic, but more challenging scenario, where it is quite possible that the algorithm will select members of the other cluster. Each of these groups consisted of 200 points as a function of the same two variables as before. These groups were made adjacent, since it was expected that a Pareto-optimal population would, in many problems, have at least some adjacent clusters.

Good agreement was achieved between the centres of each cluster,  $M_{c_{1,2}}$  and the

respective centers of each data group,  $M_{d_{1,2}}$ .

$$M_{c_1} = \begin{bmatrix} 0.0377 & -0.0295 \end{bmatrix} \quad M_{d_1} = \begin{bmatrix} 0 & 0 \end{bmatrix}$$

$$M_{c_2} = \begin{bmatrix} 3.9391 & 3.8571 \end{bmatrix} \quad M_{d_2} = \begin{bmatrix} 4 & 4 \end{bmatrix}$$

The correlation matrices of each cluster,  $C_{c_{1,2}}$ , corresponded well to those of the data,  $C_{d_{1,2}}$ , produced by the random number generator. In turn,  $C_{d_{1,2}}$  were similar to the population correlation matrices,  $C_{p_{1,2}}$ . When comparing the size of the two resulting clusters, cluster 1 had a membership of 194, whereas cluster 2 had a membership of 206 data points. Evidence of this can be seen in Figure A.3. However, the correlations were still similar to those of the sample.

$$C_{p_1} = \begin{bmatrix} 1.0000 & \\ 0.9254 & 1.0000 \end{bmatrix} \quad C_{d_1} = \begin{bmatrix} 1.0000 & \\ 0.9196 & 1.0000 \end{bmatrix} \quad C_{c_1} = \begin{bmatrix} 1.0000 & \\ 0.9046 & 1.0000 \end{bmatrix}$$

$$C_{p_2} = \begin{bmatrix} 1.0000 & \\ 0.9254 & 1.0000 \end{bmatrix} \quad C_{d_2} = \begin{bmatrix} 1.0000 & \\ 0.9253 & 1.0000 \end{bmatrix} \quad C_{c_2} = \begin{bmatrix} 1.0000 & \\ 0.9296 & 1.0000 \end{bmatrix}$$

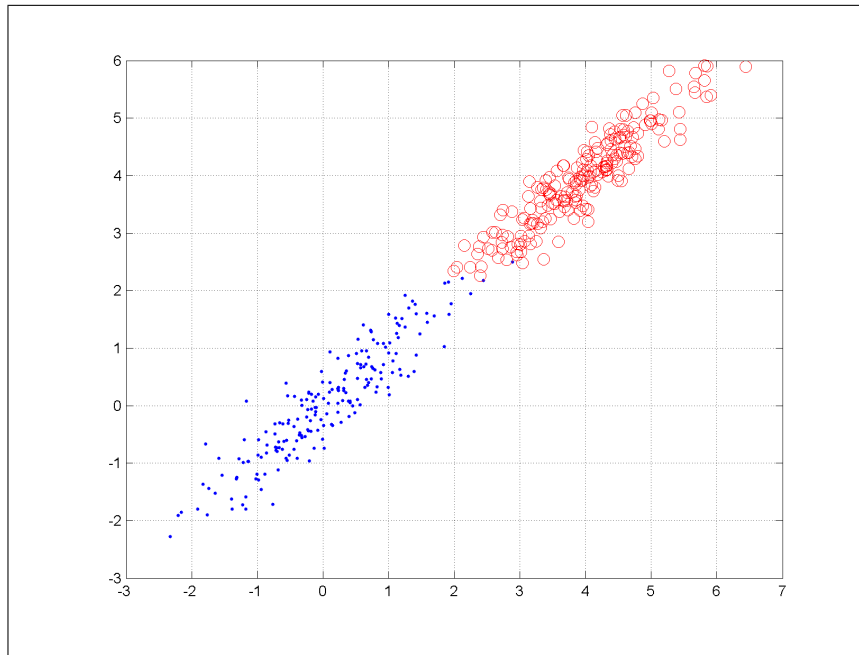


Figure A.3: Two adjacent groups of normal data.

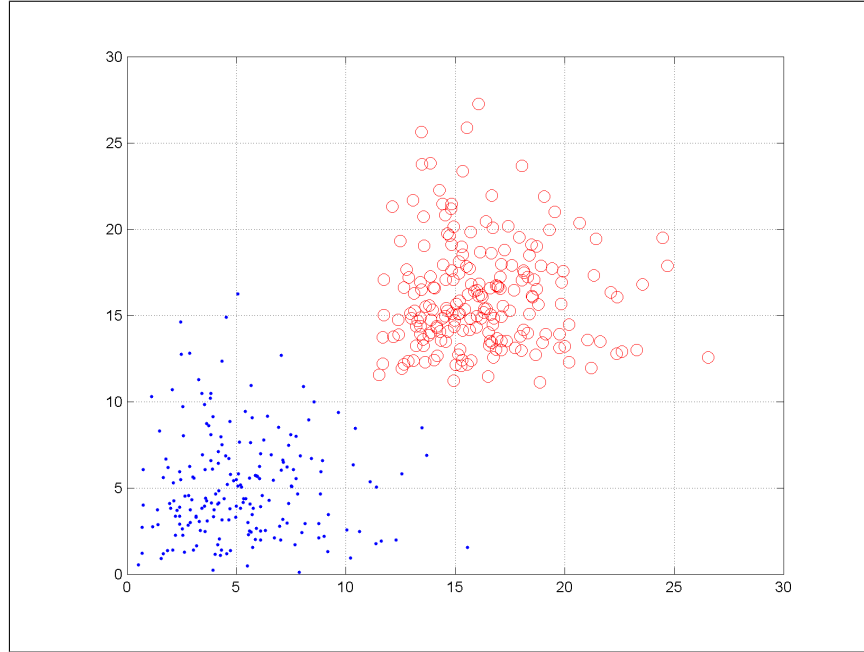


Figure A.4: Two adjacent groups of  $\chi^2$  non-normal data.

#### 4. Non-normal data: two adjacent clusters, two variables

It is quite likely that a population of Pareto-optimal solutions will be non-normal. To test one example of data from a non-normal distribution, a  $\chi^2$  distribution with 5 degrees of freedom (to give a reasonably well skewed distribution) was randomly sampled. This resulted in two adjacent data groups each of 200 points and each as a function of the same two variables as before. Good agreement was achieved between the centres of each cluster,  $M_{c_{1,2}}$  and the respective centers of each data group,  $M_{d_{1,2}}$ .

$$M_{c_1} = \begin{bmatrix} 5.1201 & 5.0623 \end{bmatrix} \quad M_{d_1} = \begin{bmatrix} 5.2102 & 5.0991 \end{bmatrix}$$

$$M_{c_2} = \begin{bmatrix} 16.226 & 15.979 \end{bmatrix} \quad M_{d_2} = \begin{bmatrix} 16.210 & 16.099 \end{bmatrix}$$

The correlation matrices of each cluster,  $C_{c_{1,2}}$ , corresponded well to those of the data,  $C_{d_{1,2}}$ , produced by the random number generator. When comparing the size of the two resulting clusters, cluster 1 had a membership of 198, whereas cluster 2 had a membership of 202 data points. Evidence of this can be seen in

Figure A.4. However, the correlations were still similar to those of the sample.

$$C_{d_1} = \begin{bmatrix} 1.0000 & \\ -0.0072 & 1.0000 \end{bmatrix} \quad C_{c_1} = \begin{bmatrix} 1.0000 & \\ -0.0348 & 1.0000 \end{bmatrix}$$

$$C_{d_2} = \begin{bmatrix} 1.0000 & \\ -0.0072 & 1.0000 \end{bmatrix} \quad C_{c_2} = \begin{bmatrix} 1.0000 & \\ 0.0177 & 1.0000 \end{bmatrix}$$

### 5. Normal data: two adjacent clusters, six variables

One of the real world optimisation problems in which it was intended to apply this algorithm consisted of six decision variables, so it was worth trying this scenario. To begin with, each of these adjacent groups consisted of 200 points as a function of the same two variables as before, as shown in the parallel coordinates plot in Figure A.5. From Table A.1 it can be seen that:

- Reasonable agreement was achieved between the respective data and cluster centres.
- Better agreement was achieved between the respective data and cluster centres when the data group size was increased from 200 to 300 points.

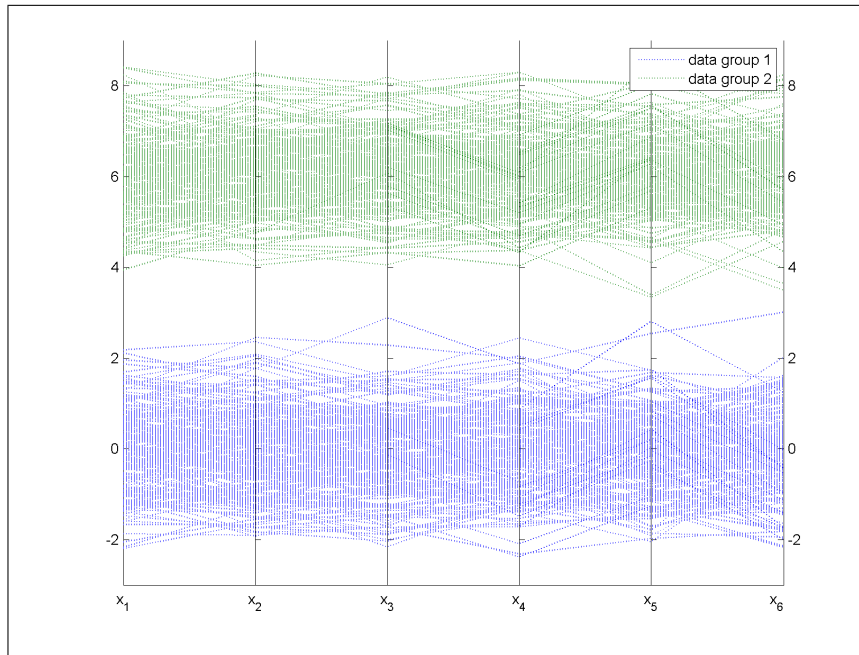


Figure A.5: Parallel coordinates plot of data groups of 200 points.



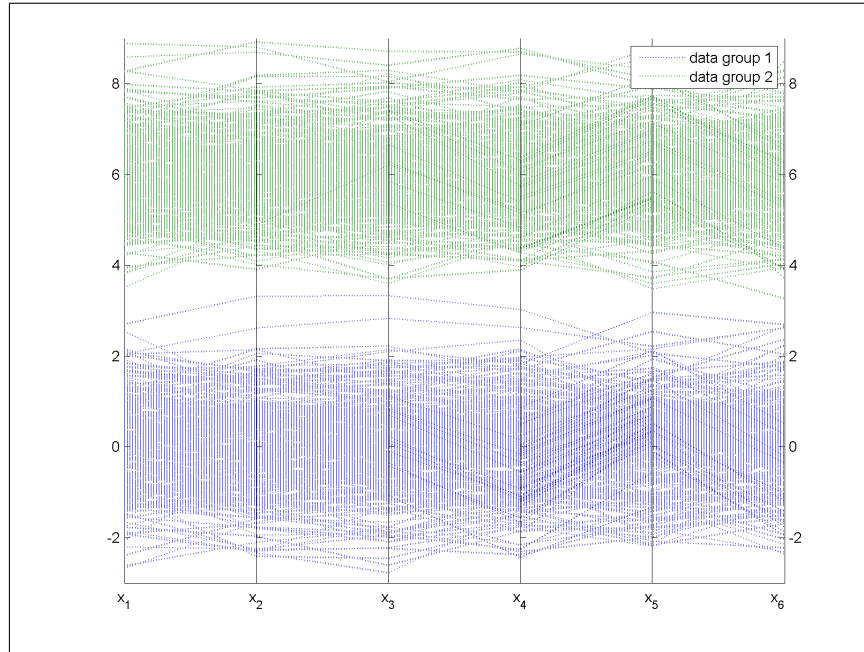


Figure A.6: Parallel coordinates plot of data groups of 300 points.

- For the data groups of 200 points, the cluster correlation matrices of each cluster corresponded well to those of the data, which, in turn, were similar to those of the population. When comparing the size of the two resulting clusters, both had memberships of 200.
- When the data group size was increased from 200 to 300 points, there was very little improvement in correspondence between the correlation matrices of the data and the corresponding clusters.

Table A.1: Comparison of centres and correlation matrices for six variable two cluster data for different data group sizes.

Cluster Property	Data Group Size	Data Source	Cluster 1						Cluster 2					
Centres	200	Data Cluster	0	0	0	0	0	0	6	6	6	6	6	6
	300	Data Cluster	-0.0388	0.1286	-0.0352	-0.2016	-0.1103	-0.1320	6.0504	6.0748	6.2160	5.9797	6.0967	6.0009
Correlation Matrices	200	Population	0	0	0	0	0	0	6	6	6	6	6	6
			0.0720	0.0202	-0.0925	-0.0406	0.0497	0.0472	5.9454	6.0558	6.0445	5.9855	6.0490	5.9223
			1						1					
			0.7791	1					0.7791	1				
			0.7112	0.8759	1				0.7112	0.8759	1			
			0.9008	0.9389	0.8493	1			0.9008	0.9389	0.8493	1		
	300	Data	0.7916	0.6592	0.7468	0.6819	1		0.7916	0.6592	0.7468	0.6819	1	
			0.6403	0.4290	0.7021	0.6039	0.6960	1	0.6403	0.4290	0.7021	0.6039	0.6960	1
		Cluster	1						1					
			0.8148	1					0.8209	1				
			0.7421	0.8944	1				0.7220	0.8821	1			
			0.9144	0.9465	0.8673	1			0.9188	0.9477	0.8520	1		
	200	Population	0.7995	0.6983	0.7557	0.6954	1		0.8149	0.7400	0.7874	0.7456	1	
			0.6692	0.5223	0.7481	0.6660	0.6869	1	0.6605	0.5308	0.7636	0.6730	0.7113	1
		Data	1						1					
			0.8113	1					0.8156	1				
			0.7385	0.8901	1				0.7177	0.8771	1			
			0.9105	0.9423	0.8630	1			0.9136	0.9419	0.8472	1		
	300	Cluster	0.7954	0.6945	0.7517	0.6921	1		0.8096	0.7348	0.7825	0.7410	1	
			0.6655	0.5191	0.7436	0.6621	0.6829	1	0.6558	0.5274	0.7590	0.6695	0.7076	1
	200	Population	1						1					
			0.7791	1					0.7791	1				
			0.7112	0.8759	1				0.7112	0.8759	1			
			0.9008	0.9389	0.8493	1			0.9008	0.9389	0.8493	1		
			0.7916	0.6592	0.7468	0.6819	1		0.7916	0.6592	0.7468	0.6819	1	
			0.6403	0.4290	0.7021	0.6039	0.6960	1	0.6403	0.4290	0.7021	0.6039	0.6960	1
	300	Data	1						1					
			0.7622	1					0.7758	1				
			0.7113	0.8517	1				0.6675	0.8610	1			
			0.8964	0.9315	0.8524	1			0.9130	0.9290	0.8095	1		
			0.8169	0.6618	0.7383	0.7103	1		0.7807	0.6417	0.7118	0.6770	1	
			0.5521	0.2646	0.6137	0.4928	0.6066	1	0.6433	0.4211	0.6737	0.6100	0.6852	1
	200	Cluster	1						1					
			0.7612	1					0.7721	1				
			0.7109	0.8487	1				0.6655	0.8555	1			
			0.8926	0.9270	0.8495	1			0.9079	0.9224	0.8049	1		
			0.8139	0.6622	0.7369	0.7095	1		0.7778	0.6402	0.7086	0.6749	1	
			0.5555	0.2743	0.6164	0.4978	0.6091	1	0.6420	0.4230	0.6727	0.6099	0.6835	1

## Appendix B

# The Comparison Operator

This Appendix provides details of some simple example problems to explain how the Comparison Operator (Fonseca and Fleming, 1998a) introduced in Section 4.3.2, operates.

### B.1 Some Examples explaining how Different Parts of the Comparison Operator work

These examples are simple two-objective, two-individual problems where both objectives are minimised. The objectives chosen relate to an automotive problem of minimising both Combustion Stability and negative Torque (maximising Torque). More extensive examples are given in the next section.

1. These are Pareto optimisation examples as shown in Figure B.1 where the priority  $r$  equals 1 so Equation (4.2) applies. Also, no goals for the objectives are given, *i.e.* the goals equal  $-\infty$ , and have equal priority.
  - Comparing individuals: A to B; A dominates B with respect to both objectives. This relates to the first part of Equation (4.2) where A does not meet the goals  $(-\infty)$ , *i.e.*  $(\mathbf{A}_r^{\widehat{\mathbf{A}}} < \mathbf{B}_r^{\widehat{\mathbf{A}}}) = 1$ . Therefore, A is preferable to B.
  - Comparing individuals: B to C which are coincident; B does not meet the goals, but B does not dominate C, since for at least one objective, B must be strictly less than C, so  $(\mathbf{B}_r^{\widehat{\mathbf{B}}} < \mathbf{C}_r^{\widehat{\mathbf{B}}}) = 0$ . Although B equals C in both objectives, *i.e.*  $(\mathbf{B}_r^{\widehat{\mathbf{B}}} = \mathbf{C}_r^{\widehat{\mathbf{B}}}) = 1$ , this is ANDed with the remainder of

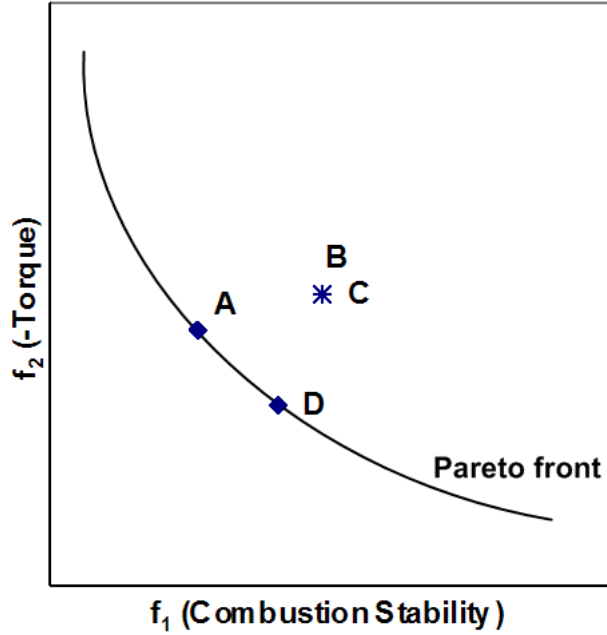


Figure B.1: Initial Pareto optimisation examples

Equation (4.2) where B does meet the goals. However, in this case B does not meet the goals and therefore the remainder of Equation (4.2) does not apply. Therefore, B is not preferable to C.

However, according to Lemma 1,  $B$  and  $C$  should be *equivalent*. So, with reference to the definition of Equivalence, Equation (4.4), B does not meet the goals, so  $(\mathbf{B}^{\mathbf{B}} = \mathbf{C}^{\mathbf{B}}) = 1$ . Also,  $(\mathbf{B}_1^{\mathbf{B}} = \mathbf{C}_1^{\mathbf{B}}) = 1$ . However, the last part does not apply as there are no higher priority objectives. So B is equivalent to C, *i.e.*  $(\mathbf{B} \equiv_{\mathbf{g}} \mathbf{C})$ .

- Comparing individuals: A to D; A does not dominate or equal D, *i.e.*  $(\mathbf{A}_r^{\mathbf{A}} \prec \mathbf{D}_r^{\mathbf{A}}) = 0$  and  $(\mathbf{A}_r^{\mathbf{A}} = \mathbf{D}_r^{\mathbf{A}}) = 0$ . So, A is not preferable to D.

2. These are constrained optimisation examples as shown in Figure B.2 where the priority  $r$  equals 2, so Equation (4.3) applies. From Equation (4.1), the functional parts,  $f_1$  in this case, of the constraint inequalities are treated as high priority objectives to be minimised until the constant parts,  $g_1$  in this case, specified as the goals, are reached. The unconstrained objectives are treated as

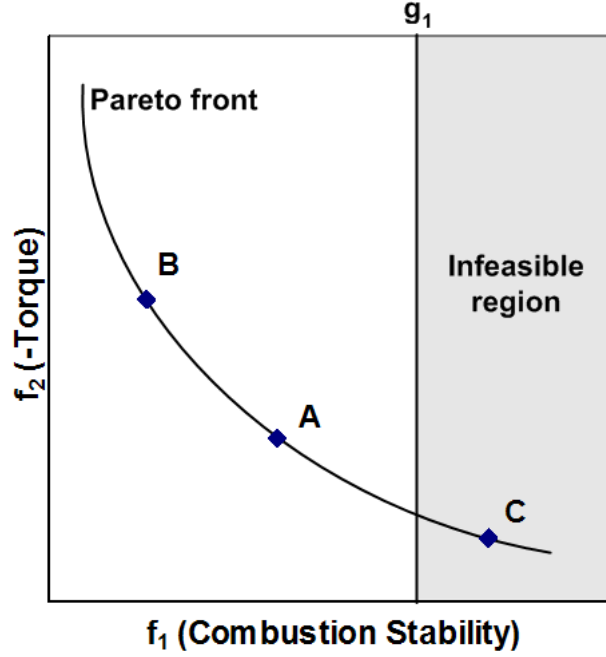


Figure B.2: Initial constrained optimisation examples

lower priority and have goals equal to  $-\infty$ . So in this case, the aim is to minimise Combustion Stability until the goal (constraint) is reached and then to minimise -Torque.

- Comparing individuals: A to B; using the convention  $A_{r,i}$  where  $m$  relates to the objective and  $r$  the priority, and considering the highest priority objective first, *i.e.* the constrained objective  $f_1$ ,  $A_{2,1}$  meets the Combustion Stability goal (or constraint)  $g_1$  and therefore the set  $\hat{A}_{2,1}^A$  is empty. This means that  $(\hat{A}_{2,1}^A = \hat{B}_{2,1}^A) = 1$ . As  $A_{2,1}$  meets the Combustion Stability goal  $g_1$ , the parts of Equation (4.3) in square brackets apply.  $B_{2,1}$  is less than  $g_1$ , so  $(\hat{B}_{2,1}^A \not\leq \hat{g}_1^A) = 0$ . However, the last part  $(\mathbf{A}_{1,\dots,r-1} \prec_{\mathbf{g}_{1,\dots,r-1}} \mathbf{B}_{1,\dots,r-1})$  does apply, and essentially means that the next priority down ( $r - 1$ ) is considered, *i.e.* the priority  $r$  equals 1 case: Equation (4.2).  $A_{1,1}$  does not meet the goal:  $-\infty$ , for this objective, and so  $(\hat{A}_{1,1}^A \prec \hat{B}_{1,1}^A) = 1$ . Therefore, A is preferable to B.
- Comparing individuals: A to C; again considering the highest priority

objective first, *i.e.* the constrained objective  $f_1$ ,  $A_{2,1}$  meets the Combustion Stability goal (or constraint)  $g_1$ . Therefore, the set  $\widehat{A_{2,1}^A}$  is empty and  $(\widehat{A_{2,1}^A} = \widehat{C_{2,1}^A}) = 1$ . In this case,  $C_{2,1}$  exceeds the Combustion Stability goal  $g_1$  and so  $(\widehat{C_{2,1}^A} \not\leq \widehat{g_1^A}) = 1$ . Therefore, A is preferable to C.

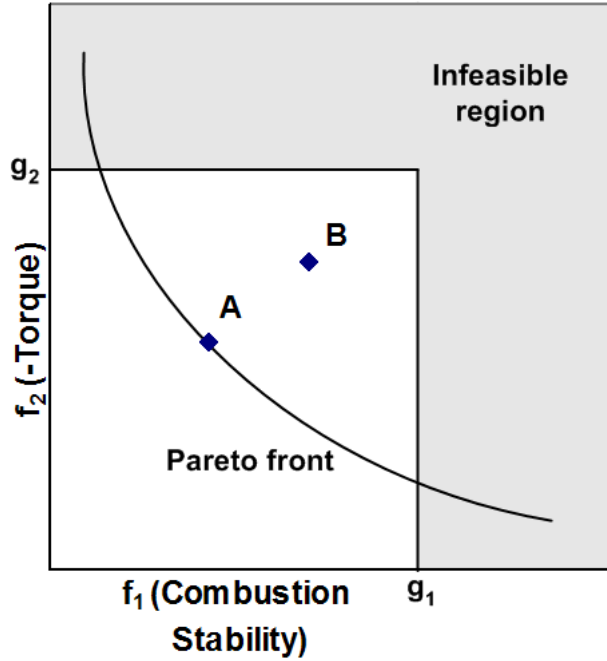


Figure B.3: Initial constraint satisfaction example

3. This is a constraint satisfaction example as shown in Figure B.3 where the priority  $r$  equals 2 so Equation (4.3) applies. All objectives are constrained and are considered in the same way as constrained optimisation, *i.e.* high priority objectives. However, as this category suggests, the aim is only to satisfy the constraints and the objectives are not minimised further once the constraints are met. In other words, there is no lower priority objective to be minimised. So, in this case, there are constraints on both -Torque and Combustion Stability. Both solutions A and B satisfy the goals (constraints) and therefore the set  $\widehat{\mathbf{A}_r^A}$  is empty. This means that  $(\widehat{\mathbf{A}_r^A} = \widehat{\mathbf{B}_r^A}) = 1$ . However, while A does meet both goals,  $(\widehat{\mathbf{B}_r^A} \not\leq \widehat{\mathbf{g}_r^A}) = 0$  and the very last part of Equation (4.3)

$(\mathbf{A}_{1,\dots,r-1} \prec_{\mathbf{g}_{1,\dots,r-1}} \mathbf{B}_{1,\dots,r-1})$  does not apply as there is no lower priority.

Therefore, A is not preferable to B.

Considering equivalence as defined in Equation (4.4), the set  $\mathbf{A}_r^{\mathbf{A}}$  is empty therefore  $(\mathbf{A}_1^{\mathbf{A}} = \mathbf{B}_1^{\mathbf{A}}) = 1$ . The next part,  $(\mathbf{A}_1^{\mathbf{A}} = \mathbf{B}_1^{\mathbf{A}})$  does not apply as  $r = 2$ .

However, the last part does. This is because where A meets the goals, B meets the goal with respect to the higher priority objectives, *i.e.*  $(\mathbf{B}_2^{\mathbf{A}} \leq \mathbf{g}_2^{\mathbf{A}}) = 1$ .

Therefore, A and B are equivalent, which is intuitive as they both satisfy the goals.

## B.2 Multi-Objective Decision Making Strategies

Fonseca and Fleming (1998a) describe several different cases with corresponding preference vectors. As in Section B.1, the examples given in this section relate to the automotive problem of minimising both Combustion Stability and negative Torque (maximising Torque) objectives referred to as  $f_1$  and  $f_2$  respectively.

### B.2.1 Pareto

All objectives have equal priority and no goals are given. The general form of the preference vector is:  $\mathbf{g} = [\mathbf{g}_1] = [(-\infty, \dots, -\infty)]$ . Worked examples have been shown in Section B.1.

### B.2.2 Lexicographic

Each objective is assigned a different priority and no goal levels are given. The general form of the preference vector is:  $\mathbf{g} = [\mathbf{g}_1, \dots, \mathbf{g}_{nobj}] = [(-\infty), \dots, (-\infty)]$ . This approach (Ben-Tal, 1980) works on the basis that the highest priority objective and goal is considered first. If a single solution results then the process terminates. However, if multiple solutions exist then the next highest priority objective and goal is considered, but without degrading the solutions from the higher priority objectives and goals. This process continues until a single solution results. The following worked example provides an explanation, see Figure B.4. For this approach,  $r = 2$  and Equation (4.3) initially applies.

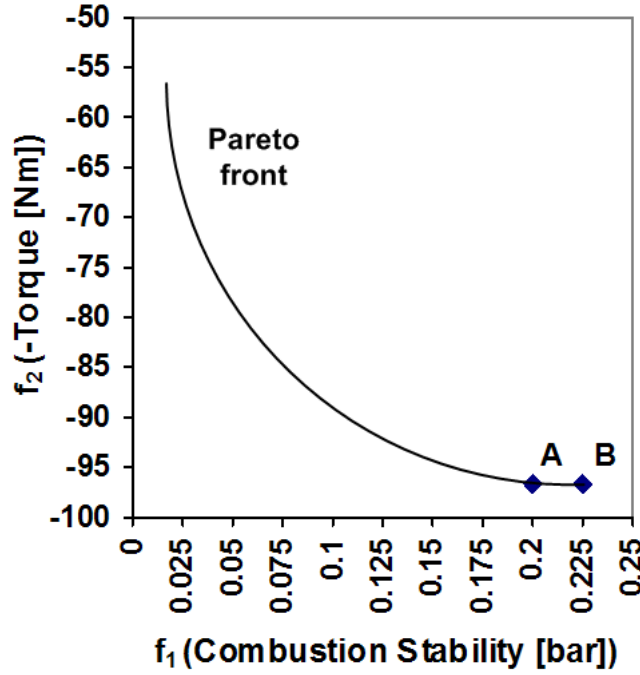


Figure B.4: Lexicographic optimisation example

### Worked examples

Comparing individuals: A to B;

$$objectives = [\mathbf{A}, \mathbf{B}] = \begin{bmatrix} 0.2 & 0.23 \\ -96.7 & -96.7 \end{bmatrix} goals = \begin{bmatrix} -\infty \\ -\infty \end{bmatrix} priorities = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$$

With respect to the highest priority constrained objective  $f_2$ , *i.e.* -Torque,

$(A_{2,2}^A = B_{2,2}^A) \cdot (B_{2,2}^A \not\leq g_2^A) = 0$ , but  $(\mathbf{A}_{1,\dots,r-1} \prec_{\mathbf{g}_{1,\dots,r-1}} \mathbf{B}_{1,\dots,r-1})$  does apply.

Therefore the next priority down ( $r - 1$ ) is considered, *i.e.* the priority  $r$  equals 1

case: Equation (4.2).  $A_{1,1}$  does not meet the goal:  $-\infty$ , for this objective, and so

$(A_{1,1}^A \prec_{\mathbf{g}} B_{1,1}^A) = 1$  and so,  $\mathbf{A} \prec_{\mathbf{g}} \mathbf{B}$ . In other words, A is equivalent to B in the (first)

highest objective Combustion Stability, but A dominates B in the lower priority

objective -Torque.



### B.2.3 Constrained Optimisation

See the constrained optimisation examples in Section B.1 for an explanation of the formulation. The general form of the preference vector is

$\mathbf{g} = [\mathbf{g}_1, \mathbf{g}_2] = [(-\infty, \dots, -\infty), (g_{2,1}, \dots, g_{2,nobj_c})]$ , which, for these examples translates to:  $\mathbf{g} = [\mathbf{g}_1, \mathbf{g}_2] = [(0.15), (-\infty)]$ , see Figure B.5. That is,  $r = 2$  and Equation (4.3) applies.

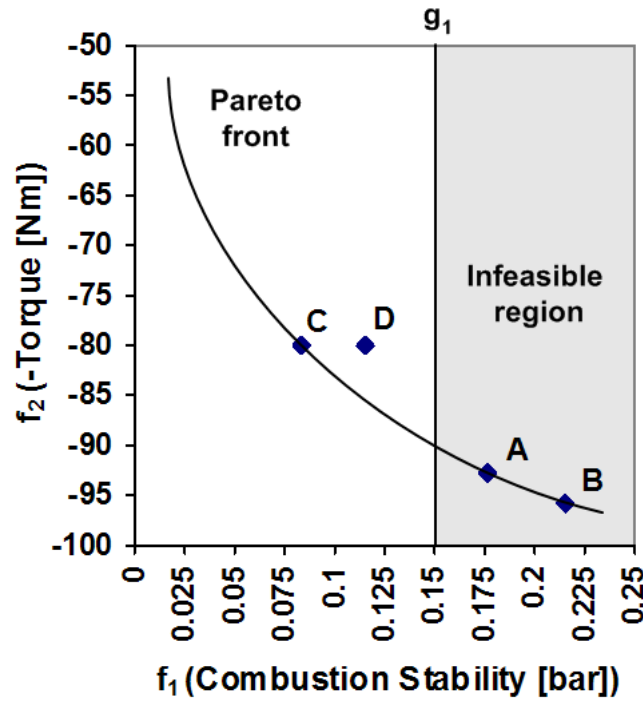


Figure B.5: Further constrained optimisation examples

#### Worked examples

Comparing individuals: A to B;

$$objectives = [\mathbf{A}, \mathbf{B}] = \begin{bmatrix} 0.18 & 0.22 \\ -93 & -96 \end{bmatrix} \quad goals = \begin{bmatrix} 0.15 \\ -\infty \end{bmatrix} \quad priorities = \begin{bmatrix} 2 \\ 1 \end{bmatrix}$$

With respect to the highest priority constrained objective  $f_1$ ,  $A_{2,1} > g_1$ , *i.e.*

$A_{2,1} > 0.15$ . So,  $(A_{2,1}^A < B_{2,1}^A) = 1$  and  $\mathbf{A} \prec_{\mathbf{g}} \mathbf{B}$ . This is because A gets closer than B to

the highest priority objective, which is the Combustion Stability constraint in this case.

Comparing individuals: C to D;

$$objectives = [\mathbf{C}, \mathbf{D}] = \begin{bmatrix} 0.08 & 0.12 \\ -80 & -80 \end{bmatrix} goals = \begin{bmatrix} 0.15 \\ -\infty \end{bmatrix} priorities = \begin{bmatrix} 2 \\ 1 \end{bmatrix}$$

With respect to the highest priority constrained objective  $f_1$ ,  $C_{2,1} < g_1$ , so  $C_{2,1}^C = \{\}$  and  $(C_{2,1}^C = D_{2,1}^C) = 1$ . As  $(D_{2,1}^C \not\leq g_1^C) = 0$ , i.e.  $C_{2,1} < 0.15$ , the next priority down ( $r = 1$ ) is considered.  $C_{1,1}$  does not meet the goal:  $-\infty$ , for this objective, but C does not dominate D in the lower priority objective and so  $(C_{1,1}^C \prec B_{1,1}^C) = 0$ . Therefore, C is not preferable to D.

Considering equivalence as defined in Equation (4.4), the set  $\mathbf{C}_r^C$  is empty therefore  $(\mathbf{C}^C = \mathbf{D}^C) = 1$ . The next part,  $(\mathbf{A}_1^A = \mathbf{B}_1^A) = 1$ . However, the last part does not apply as  $r = 1$ . Therefore, C and D are equivalent with respect to the lower priority objective,  $f_1$ .

### B.2.4 Constraint Satisfaction

All constraints are treated as in constrained optimisation, but there is no low priority objective to be optimised. In other words, all objectives are constrained and the only aim is to satisfy all constraints.

The general form of the preference vector is:  $\mathbf{g} = [\{\}, \mathbf{g}_2] = [(g_{2,1}, \dots, g_{2,nobj})]$ , which for these examples translates to:  $\mathbf{g} = [\{\}, \mathbf{g}_2] = [(0.15, -70)]$ , see Figure B.6. That is,  $r = 2$  and Equation (4.3) applies.

### Worked examples

1. Comparing individuals: A to B;

$$objectives = [\mathbf{A}, \mathbf{B}] = \begin{bmatrix} 0.09 & 0.18 \\ -80.4 & -92.8 \end{bmatrix} goals = \begin{bmatrix} 0.15 \\ -70 \end{bmatrix} priorities = \begin{bmatrix} 2 \\ 2 \end{bmatrix}$$

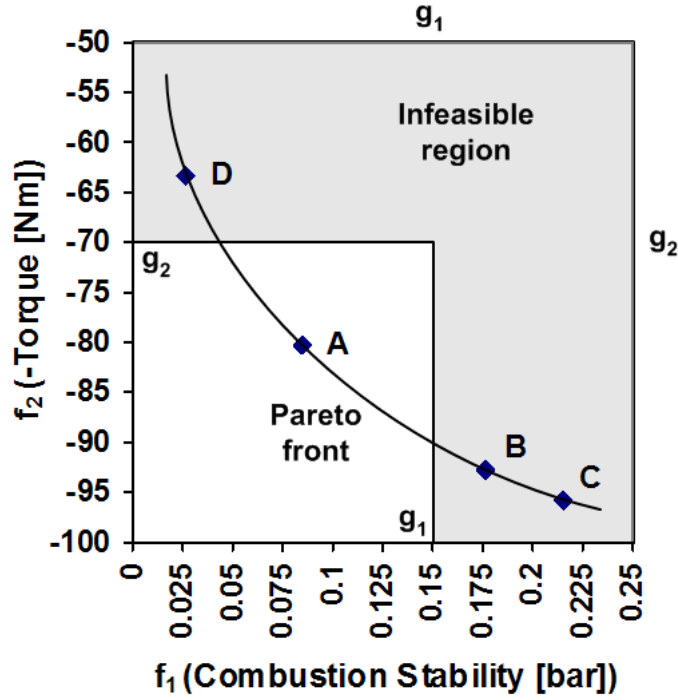


Figure B.6: Further constraint satisfaction examples

A meets both goals (constraints), *i.e.*  $\mathbf{A}^{\hat{}} = \{\}$ , so  $(\mathbf{A}_r^{\hat{}} = \mathbf{B}_r^{\hat{}}) = 1$  and  $(B_{2,1}^{\hat{}} \not\leq g_1^{\hat{}}) = 1 \quad \therefore \mathbf{A} \prec_{\mathbf{g}} \mathbf{B}$ .

2. Comparing individuals: B to C;

$$\text{objectives} = [\mathbf{B}, \mathbf{C}] = \begin{bmatrix} 0.18 & 0.22 \\ -92.8 & -95.9 \end{bmatrix} \quad \text{goals} = \begin{bmatrix} 0.15 \\ -70 \end{bmatrix} \quad \text{priorities} = \begin{bmatrix} 2 \\ 2 \end{bmatrix}$$

Considering first where B does not meet the goal(s) (constraint(s)), which in this example is  $B_{2,1} > g_1$ , *i.e.*  $B_{2,1} > 0.15 \quad \therefore B_{2,1}^B \prec C_{2,1}^B$  and  $\mathbf{B} \prec_{\mathbf{g}} \mathbf{C}$ . This is because where B exceeds the constraints it gets closer than C to satisfying them.

3. Comparing individuals: B to D;

$$\text{objectives} = [\mathbf{B}, \mathbf{D}] = \begin{bmatrix} 0.18 & 0.03 \\ -92.8 & -63.2 \end{bmatrix} \quad \text{goals} = \begin{bmatrix} 0.15 \\ -70 \end{bmatrix} \quad \text{priorities} = \begin{bmatrix} 2 \\ 2 \end{bmatrix}$$

B does not meet goal (constraint)  $g_1$ , *i.e.*  $B_{2,1} > 0.15$ , but  $(B_{2,1}^B \prec D_{2,1}^B) = 0$

and  $(B_{2,1}^B = D_{2,1}^B) = 0$ . Likewise, B does not meet goal (constraint)  $g_2$ , *i.e.*  $B_{2,2} > -70$ , but  $(B_{2,2}^B < D_{2,2}^B) = 0$  and  $(B_{2,2}^B = D_{2,2}^B) = 0$ .  $\therefore \mathbf{B} \not\prec_{\mathbf{g}} \mathbf{D}$ . This is because B and D fail different constraints, so their performance relative to the constraints cannot be compared.

### B.2.5 Goal Programming(1)

One formulation (Hwang and Masud, 1979) consists of trying to meet the goals simultaneously in a similar way to lexicographic optimisation. The general form of the preference vector is:  $\mathbf{g} = [\mathbf{g}_1, \dots, \mathbf{g}_{nobj}] = [(g_{1,1}), \dots, (g_{nobj,1})]$ , which for these examples translates to:  $\mathbf{g} = [\mathbf{g}_1, \mathbf{g}_2] = [(0.15), (-70)]$ , see Figure B.7. That is,  $r = 2$  and Equation (4.3) applies.

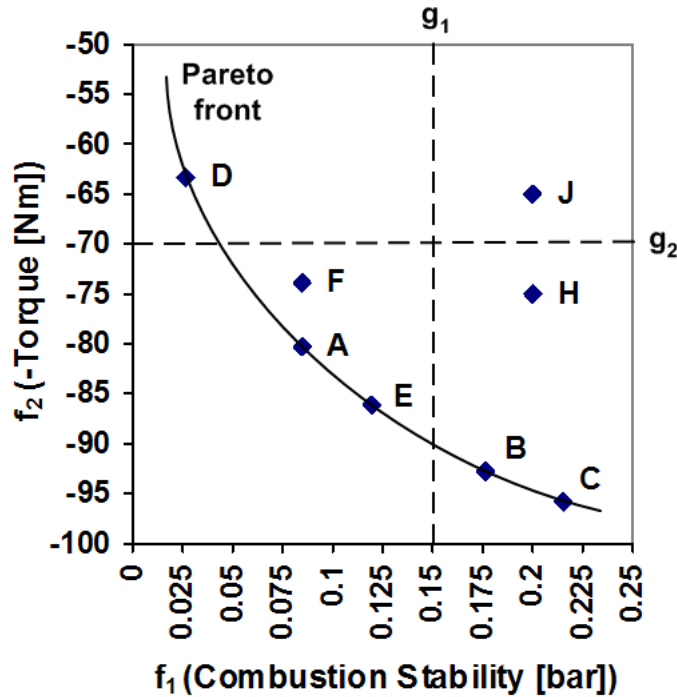


Figure B.7: Goal programming examples

## Worked examples

1. Comparing individuals: A to E;

$$objectives = [\mathbf{A}, \mathbf{E}] = \begin{bmatrix} 0.09 & 0.12 \\ -80.4 & -86.2 \end{bmatrix} goals = \begin{bmatrix} 0.15 \\ -70 \end{bmatrix} priorities = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$$

A meets both goals, *i.e.*  $\widehat{\mathbf{A}_r} = \{\}$ , so  $(\widehat{\mathbf{A}_r} = \widehat{\mathbf{E}_r}) = 1$ . Considering the highest priority objective  $f_2$  (-Torque),  $(\widehat{E_{2,2}} \not\leq \widehat{g_2}) = 0$ , but

$(\mathbf{A}_{1,\dots,r-1} \prec_{\mathbf{g}_{1,\dots,r-1}} \mathbf{E}_{1,\dots,r-1})$  applies. This means the next priority down is considered, *i.e.*  $r = 1$  and Equation (4.2) applies.  $(\widehat{A_{1,1}} \prec \widehat{E_{1,1}}) = 1 \quad \therefore \mathbf{A} \prec_{\mathbf{g}} \mathbf{E}$ . This is because A gets closer to the higher priority goal  $g_2$ .

2. Comparing individuals: A to B;

$$objectives = [\mathbf{A}, \mathbf{B}] = \begin{bmatrix} 0.09 & 0.18 \\ -80.4 & -92.8 \end{bmatrix} goals = \begin{bmatrix} 0.15 \\ -70 \end{bmatrix} priorities = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$$

A meets both goals (constraints), *i.e.*  $\widehat{\mathbf{A}_r} = \{\}$ , so  $(\widehat{\mathbf{A}_r} = \widehat{\mathbf{B}_r}) = 1$ .

Considering the highest priority objective  $f_2$  (-Torque),  $(\widehat{B_{2,2}} \not\leq \widehat{g_2}) = 0$ , but

$(\mathbf{A}_{1,\dots,r-1} \prec_{\mathbf{g}_{1,\dots,r-1}} \mathbf{B}_{1,\dots,r-1})$  applies. This means the next priority down is considered, *i.e.*  $r = 1$  and Equation (4.2) applies.  $(\widehat{B_{1,1}} \not\leq \widehat{g_1}) = 1 \quad \therefore \mathbf{A} \prec_{\mathbf{g}} \mathbf{B}$ .

This is because A meets both goals but B fails goal  $g_1$ .

3. Comparing individuals: B to C;

$$objectives = [\mathbf{B}, \mathbf{C}] = \begin{bmatrix} 0.18 & 0.22 \\ -92.8 & -95.9 \end{bmatrix} goals = \begin{bmatrix} 0.15 \\ -70 \end{bmatrix} priorities = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$$

Considering first the highest priority objective  $f_2$  (-Torque),  $\widehat{B_{2,2}} = \{\}$ , so

$(\widehat{B_{2,2}} = \widehat{C_{2,2}}) = 1$ .  $(\widehat{C_{2,2}} \not\leq \widehat{g_2}) = 0$ , but  $(\mathbf{B}_{1,\dots,r-1} \prec_{\mathbf{g}_{1,\dots,r-1}} \mathbf{C}_{1,\dots,r-1})$  applies.

This means the next priority down is considered, *i.e.*  $r = 1$  and Equation (4.2)

applies.  $(\widehat{B_{1,1}} \prec \widehat{C_{1,1}}) = 1 \quad \therefore \mathbf{B} \prec_{\mathbf{g}} \mathbf{C}$ . This is because while both B and C meet goal  $g_2$  and fail to meet goal  $g_1$ , B gets closer than C to satisfying goal  $g_1$ .

4. Comparing individuals: B to D;

$$objectives = [\mathbf{B}, \mathbf{D}] = \begin{bmatrix} 0.18 & 0.03 \\ -92.8 & -63.2 \end{bmatrix} goals = \begin{bmatrix} 0.15 \\ -70 \end{bmatrix} priorities = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$$

Considering first the highest priority objective  $f_2$  (-Torque),  $B_{2,2}^B = \{\}$ , so  $(B_{2,2}^B = D_{2,2}^B) = 1$ .  $(D_{2,2}^B \not\leq g_2^B) = 1 \quad \therefore \mathbf{B} \prec_{\mathbf{g}} \mathbf{D}$ . This is because B meets, but D fails to meet, the higher priority goal  $g_2$ .

5. Comparing individuals: A to F;

$$objectives = [\mathbf{A}, \mathbf{F}] = \begin{bmatrix} 0.09 & 0.09 \\ -80.4 & -74 \end{bmatrix} goals = \begin{bmatrix} 0.15 \\ -70 \end{bmatrix} priorities = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$$

A meets both goals, *i.e.*  $\mathbf{A}_{\hat{r}}^{\mathbf{A}} = \{\}$ , so  $(\mathbf{A}_{\hat{r}}^{\mathbf{A}} = \mathbf{F}_{\hat{r}}^{\mathbf{A}}) = 1$ . Considering first the highest priority objective  $f_2$  (-Torque),  $(F_{2,2}^A \not\leq \mathbf{g}_2^A) = 0$ , but  $(\mathbf{A}_{1,\dots,r-1} \prec_{\mathbf{g}_{1,\dots,r-1}} \mathbf{F}_{1,\dots,r-1})$  applies. This means the next priority down is considered, *i.e.*  $r = 1$  and Equation(4.2) applies.  $(A_{1,1}^A \prec F_{1,1}^A) = 0$ , since for  $A_{1,1}$  to dominate  $F_{1,1}$ ,  $A_{1,1}$  must be strictly less than  $F_{1,1}$ .  $\therefore \mathbf{A} \not\prec_{\mathbf{g}} \mathbf{F}$ .

However, according to Lemma 1,  $A$  and  $F$  should be *equivalent*. So, with reference to the definition of Equivalence, Equation (4.4), A meets both goals, *i.e.*  $\mathbf{A}_{\hat{r}}^{\mathbf{A}} = \{\}$ , so  $(\mathbf{A}_{\hat{r}}^{\mathbf{A}} = \mathbf{F}_{\hat{r}}^{\mathbf{A}}) = 1$ . Also,  $(\mathbf{A}_1^A = \mathbf{F}_1^A) = 1$ , and  $(\mathbf{F}_2^A \leq \mathbf{g}_2^A) = 1$ , so  $(\mathbf{A} \equiv_{\mathbf{g}} \mathbf{F})$ .

### B.2.6 Goal Programming(2)

Another formulation attempts to meet all goals simultaneously and be Pareto optimal. The general form of the preference vector is:  $\mathbf{g} = [\mathbf{g}_1] = [(g_{1,1}, \dots, g_{1,nobj})]$ , which for these examples translates to:  $\mathbf{g} = [\mathbf{g}_1] = [(0.15, -70)]$ , see Figure B.7. That is,  $r = 1$  and Equation(4.2) applies.

## Worked examples

1. Comparing individuals: A to E;

$$objectives = [\mathbf{A}, \mathbf{E}] = \begin{bmatrix} 0.09 & 0.12 \\ -80.4 & -86.2 \end{bmatrix} goals = \begin{bmatrix} 0.15 \\ -70 \end{bmatrix} priorities = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

A meets both goals, *i.e.*  $\mathbf{A}_r^{\mathbf{A}} = \{\}$ , so  $(\mathbf{A}_r^{\mathbf{A}} = \mathbf{E}_r^{\mathbf{A}}) = 1$ .  $(\mathbf{E}_r^{\mathbf{A}} \not\leq \mathbf{g}_r^{\mathbf{A}}) = 0$ , and

$(\mathbf{A}_{1,\dots,r-1} \prec_{\mathbf{g}_{1,\dots,r-1}} \mathbf{E}_{1,\dots,r-1})$  does not apply as there is no lower priority.

$\therefore \mathbf{A} \not\prec_{\mathbf{g}} \mathbf{E}$ . This is because both A and E satisfy both equal priority goals and do not dominate each other.

2. Comparing individuals: A to B;

$$objectives = [\mathbf{A}, \mathbf{B}] = \begin{bmatrix} 0.09 & 0.18 \\ -80.4 & -92.8 \end{bmatrix} goals = \begin{bmatrix} 0.15 \\ -70 \end{bmatrix} priorities = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

A meets both goals (constraints), *i.e.*  $\mathbf{A}_r^{\mathbf{A}} = \{\}$ , so  $(\mathbf{A}_r^{\mathbf{A}} = \mathbf{B}_r^{\mathbf{A}}) = 1$ .

$(B_{1,1}^{\mathbf{A}} \not\leq g_{1,1}^{\mathbf{A}}) = 1 \quad \therefore \mathbf{A} \prec_{\mathbf{g}} \mathbf{B}$ . This is because A meets both goals but B fails to meet goal  $g_1$ .

3. Comparing individuals: B to C;

$$objectives = [\mathbf{B}, \mathbf{C}] = \begin{bmatrix} 0.18 & 0.22 \\ -92.8 & -95.9 \end{bmatrix} goals = \begin{bmatrix} 0.15 \\ -70 \end{bmatrix} priorities = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

Considering first where B fails to meet the goals, *i.e.*  $B_{1,1} > g_1$ ,

$(B_{1,1}^{\mathbf{B}} \not\leq C_{1,1}^{\mathbf{B}}) = 1 \quad \therefore \mathbf{B} \prec_{\mathbf{g}} \mathbf{C}$ . This is because where B fails to meet the goals, *i.e.*  $g_1$ , it gets closer than C to satisfying goal  $g_1$ .

4. Comparing individuals: B to D;

$$objectives = [\mathbf{B}, \mathbf{D}] = \begin{bmatrix} 0.18 & 0.03 \\ -92.8 & -63.2 \end{bmatrix} goals = \begin{bmatrix} 0.15 \\ -70 \end{bmatrix} priorities = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

Considering first where B fails to meet the goals, *i.e.*  $B_{1,1} > g_1$ ,

$(B_{1,1}^{\mathbf{B}} \not\leq D_{1,1}^{\mathbf{B}}) = 0$  and  $(B_{1,1}^{\mathbf{B}} = D_{1,1}^{\mathbf{B}}) = 0 \quad \therefore \mathbf{B} \not\prec_{\mathbf{g}} \mathbf{D}$ . This is because while B

and D both fail to meet one of the goals, they fail different goals, which both have equal priority.

5. Comparing individuals: H to J;

$$objectives = [\mathbf{H}, \mathbf{J}] = \begin{bmatrix} 0.2 & 0.2 \\ -75 & -65 \end{bmatrix} goals = \begin{bmatrix} 0.15 \\ -70 \end{bmatrix} priorities = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

Considering first where H fails to meet the goals, *i.e.*  $H_{1,1} > g_1$ ,

$(\widehat{H}_{1,1}^H \prec \widehat{J}_{1,1}^J) = 0$ , but  $(\widehat{H}_{1,1}^H = \widehat{J}_{1,1}^H) = 1$ . H meets the -Torque goal, *i.e.*

$H_{1,2} < g_2$  and  $(\widehat{J}_{1,2}^H \not\leq \widehat{g}_2^H) = 1$ .  $\therefore \mathbf{H} \prec_{\mathbf{g}} \mathbf{J}$ . This is because while H and J both

fail to meet one of the goals (by the same extent), H meets the other goal,

whereas J does not and so H is preferable to J.



## Appendix C

# Application of Savitzky-Golay smoothing to combustion data

This is an Appendix to Section 6.5 and explains how the NMEP combustion data, used in some of the objective functions, has been smoothed using the Savitzky-Golay (SG) algorithm (Savitzky and Golay, 1964) with a detailed explanation provided in Press *et al.* (1992, pp. 650).

Let  $H$  denote the set of observed combustion events and  $x_{ce}$  the Net Mean Effective Pressure (NMEP) (Ferguson, 1986) for the corresponding  $ce^{th}$  event, then:

$$H = \{x_i \in \mathbb{R}\}_{ce=1}^N \quad (\text{C.1})$$

It should be noted that  $H$  is implicitly considered an ordered set as it is assumed that the  $x_{ce}$  are written as an ordered sequence. In general, the set  $H$  may contain non-firing or aberrant combustion events. It is therefore convenient to define a subset  $F$  of  $H$  ( $F \subseteq H$ ), whose elements exclude misfire or other aberrant combustion data.  $F$  is defined by:

$$F = \{(x_{ce} \in \mathbb{R}) \wedge (x_{ce} > nm)\} \quad (\text{C.2})$$

In Equation C.2,  $nm$  is the *NMEP threshold* parameter, above which an event is defined to have normal combustion. Further, let  $x_f$  denote the individual members of  $F$  ( $x_f \in F$ ). If  $s \in [1, N] \wedge i \in \mathbb{N}$  denotes the event index at which combustion first occurs, this suggests  $x_s$  is the first member of  $F$ ; *i.e.* the NMEP achieved at first fire.

Therefore,  $s$  is the first event for which  $x_{ce} > nm$ .

It is also necessary to define  $W$  as the set:

$$W = \{x_{ce} \in \mathbb{R}\}_{i=s}^N \quad (\text{C.3})$$

That is,  $W$  is the set of NMEP values for all combustion events from first fire onwards. Clearly, by definition,  $W \subset H$ . Similarly,  $x_w$  denotes the individual members of  $W$  ( $x_w \in W$ ).

In addition, let  $y_{ce} = SG(x_{ce}, n_l, n_r, ft)$  (or  $y_{ce} = SG(n_l, n_r, ft)$  if the variable to be filtered is known) denote the result of applying a  $ft^{th}$  order Savitzky-Golay filter of window size  $n_l + n_r + 1$ , where  $n_l$  is the number of samples to the left of the  $ce^{th}$  sampling instance and  $n_r$  is the corresponding number of samples to the right, of the  $x_{ce}$  data.

Figure C.1 illustrates the application of a  $SG(x_{ce}, 4, 4, 2)$  filter to after-start NMEP data obtained from a typical engine start test run. It can be seen that the data are significantly smoothed without the need to resort to formulating an explicit explanatory model; the smoothed data being considered to represent a trend-line through the data. As a result, the deviations between the smoothed and raw data at

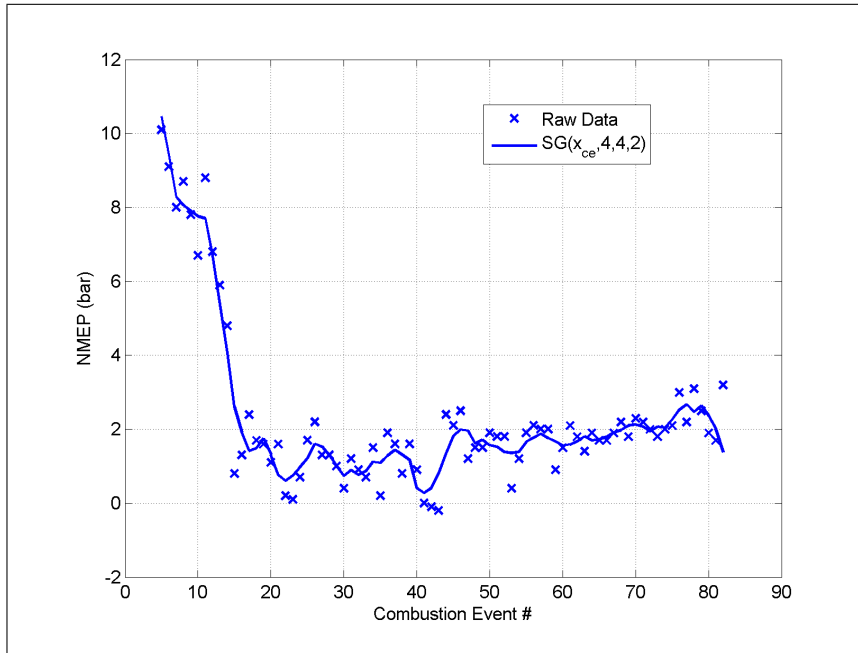


Figure C.1: Application of a  $2^{nd}$  order SG-filter (symmetrical window and window width) to a typical after-start NMEP trace.

any instant indicate the consistency or quality of the combustion process.

The set of squared NMEP deviations,  $DN$  can then be defined as:

$$DN = \{x_w, y_w \in \mathbb{R} \wedge d_w^2 = (y_w - x_w)^2\}_{w=s}^N \quad (\text{C.4})$$

where  $d_w$  is the NMEP deviation for an individual combustion event.

Figure C.2 illustrates the effect of varying the SG-filter parameters on a typical NMEP data trace; pane a) displays the smoothed response and pane b) the corresponding  $d_{ce}$ . As expected, the  $d_{ce}$  plot depends on various filter parameterisations.

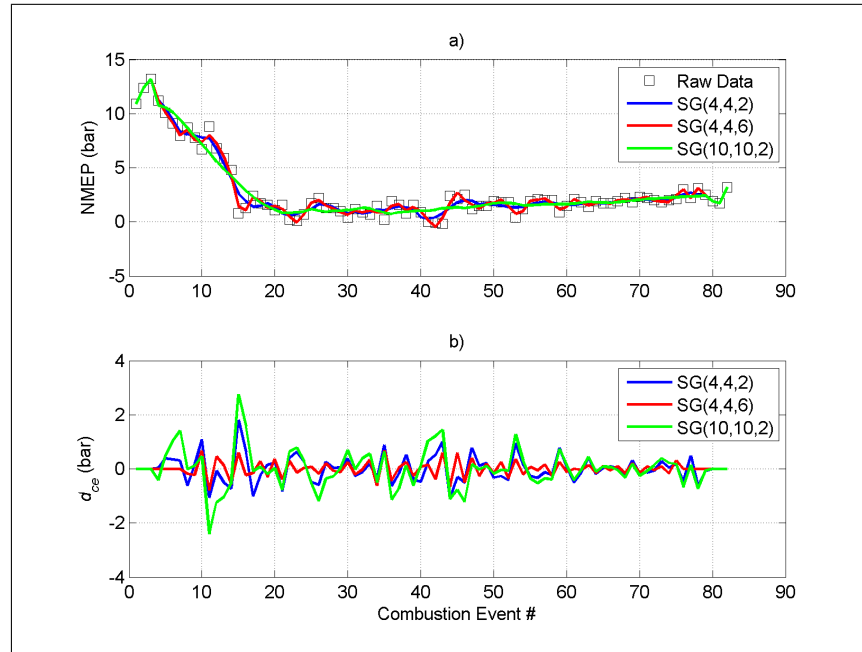


Figure C.2: Effect of SG filter parameters on the smoothed NMEP response. Pane a) shows the raw data and corresponding smoothed data. Pane b) plots the corresponding NMEP deviations. Increasing the order of the polynomial, for a fixed window width, decreases the smoothing, while increasing the window width, for a fixed polynomial order, increases the smoothing.



## Appendix D

# Mathematical Notation for Clustering Verification Rules

This notation applies to the Cluster verification Rules as defined in Section 4.4.3, and which are used to make cluster comparisons. It not only provides a compact representation, but also forms part of a requirements specification for developing automated analysis software.

### D.1 A Compact Notation for Cluster Analyses used in CR1 and CR2

We will use the following notation to describe a clustering analysis:

$$V_k^{obj}(\gamma, lr, cs, \varphi, \tau) = [\xi_1, \xi_2, \dots, \xi_s] \quad (D.1)$$

where  $k$  denotes the  $k^{th}$  sub-sample, with  $k = 1$  representing the reference sample,  $obj$  the number of objectives,  $\gamma$  the sub-sample size,  $lr$  is the learning rate,  $cs$  the initial number of clusters,  $\varphi$  the maximum number of iterations and  $\tau$  the convergence tolerance. The purpose of this notation is to completely define the clustering algorithm configuration and is currently designed for the k\*-Means clustering algorithm only. Similar notation could be subsequently developed for other clustering algorithms.

The parameters  $\xi_j \in \mathbb{N}$  denote the number of clusters resulting from each run of the algorithm. These are written from left to right in terms of frequency of occurrence

and ascending number of clusters. Thus  $V_k^{nobj}(\gamma, lr, cs, \varphi, \tau) = [2, 2, 3]$  defines that the most frequently occurring solution had two clusters, as did the next most common solution. However, the two solutions are not the same and will have different centres, correlation matrices or number of solutions per cluster. The least frequent solution resulted in three clusters. In the scenario where two solutions, *e.g.* those with two clusters, occur equally as frequently, then brackets will be used to identify them, *e.g.*  $V_k^{nobj}(\gamma, lr, cs, \varphi, \tau) = [(2, 2), 3]$ .

k\*-Means is a stochastic clustering algorithm and may therefore return different  ${}_k\xi_j$  for the same parameter settings. Consequently, k\*-Means needs to be run several times to allow discovery of potential multiple solutions.

It may be of interest to the decision maker (DM) to explore solutions corresponding to multiple  ${}_k\xi_j$ , especially in problems where little prior knowledge exists. For example, if  $V_k^{nobj}(\gamma, lr, cs, \varphi, \tau) = [2, 2, 3]$  the DM may want to look at  ${}_k\xi_1 = {}_k\xi_2 = 2$ . To differentiate between these two results, an additional index,  $z$ , is used to represent different solutions with the same number of resulting clusters leading to the notation,  $V_{kz}^{nobj}$ .

## D.2 Comparison of Cluster Centres and Correlations in CR3

### D.2.1 The Consistent Cluster Data Membership Condition

If  $D_k$  denotes the (Pareto-optimal population, or POP) solutions for the  $k^{th}$  sub-sample, then  ${}_kD_i = \{{}_k\mathbf{x}_{ij} \in \mathbb{R}^{nobj}\}_{j=1}^{n_i}$  signifies the set of  ${}_kn_i$ ,  $i \in [1, nc]$  solutions ( ${}_k\mathbf{x}_{ij}$ ) associated with the  $i^{th}$  cluster at sub-sample  $k$ , then  $D_k = \bigcup_{i=1}^{nc} {}_kD_i$ , where  $nc$  is the number of clusters.

The condition  $\{{}_kD \subset {}_1D_i\}_{i=1}^{nc}$  must necessarily apply at each sub-sampled clustering operation. This criterion is referred to as the *consistent cluster data membership condition*. We write  $\{{}_kD \subset {}_1D_i\}_{i=1}^{nc}$  to indicate the condition is satisfied and  $\{{}_kD \not\subset {}_1D_i\}_{i=1}^{nc}$  to indicate that it is not.

### D.2.2 Introduction of *vec* and *vech* Operators.

A compact vector representation is provided by the *vec* and *vech* operators (Searle, 1982). For example, let the matrix

$$\mathbf{A} = \begin{bmatrix} 1 & 3 \\ 2 & 4 \end{bmatrix}$$

Then the operator *vec*( $\bullet$ ) denotes the process of stacking the individual column vectors to comprise the matrix, *i.e.*

$$\text{vec}(\mathbf{A}) = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix}$$

Similarly, the *vech*( $\bullet$ ) operator denotes the process of stacking on or below the leading diagonal, as follows:

$$\text{vech}(\mathbf{A}) = \begin{bmatrix} 1 \\ 2 \\ 4 \end{bmatrix}$$

### D.2.3 Application of *vec* and *vech* Operators to Cluster Comparisons.

These operators can form the basis of cluster centre and correlation matrix comparisons. For example, if the cluster resulting from CR1 that is chosen for sub-sampling, is defined as the *reference* cluster, then the matrix of reference cluster centres,  $\mathbf{M}_1$  can be defined as:

$$\mathbf{M}_1 = \begin{bmatrix} {}_1\mathbf{m}_1^T \\ {}_1\mathbf{m}_2^T \\ \vdots \\ {}_1\mathbf{m}_{nc}^T \end{bmatrix} \in \mathbb{R}^{f \times nobj} \quad (\text{D.2})$$

where  ${}_1\mathbf{m}_i^T \in \mathbb{R}^{1 \times nobj}, \forall i \in \mathbb{N} \wedge i \in [1, nc];$ , *i.e.* the matrix formed by stacking the *nc*-row vectors, with *nc* clusters and *nobj* objectives.

If  $\mathbf{M}_k$  represents the corresponding matrix of cluster centres for the  $k^{th}$  sub-sample, then:

$$\mathbf{M}_k = \begin{bmatrix} {}_k\mathbf{m}_1^T \\ {}_k\mathbf{m}_2^T \\ \vdots \\ {}_k\mathbf{m}_f^T \end{bmatrix} \in \mathbb{R}^{f \times d} \quad (\text{D.3})$$

Further, if  ${}_k m_{ij}$  denotes the  $j^{th}$  element of  ${}_k \mathbf{m}_i^T$ , then the matrix  $\mathbf{N}_k$  can be defined as:

$$\mathbf{N}_k = \begin{bmatrix} {}_k\mathbf{m}_{11}^{-1} & {}_k\mathbf{m}_{12}^{-1} & \cdots & {}_k\mathbf{m}_{1ncobj}^{-1} \\ {}_k\mathbf{m}_{21}^{-1} & {}_k\mathbf{m}_{22}^{-1} & \cdots & {}_k\mathbf{m}_{2ncobj}^{-1} \\ \vdots & \vdots & & \vdots \\ {}_k\mathbf{m}_{f1}^{-1} & {}_k\mathbf{m}_{f2}^{-1} & \cdots & {}_k\mathbf{m}_{fncobj}^{-1} \end{bmatrix} \quad (\text{D.4})$$

A compact approach for comparing corresponding cluster centres between the reference cluster,  $\mathbf{M}_1$ , and the sub-sampled clusters,  $\mathbf{M}_k$  is to determine the largest absolute normalised difference,  $\Omega_k$ , between them and is calculated as follows:

$$\Omega_k = \max | (vec(\mathbf{M}_1^T) - vec(\mathbf{M}_k^T)) \circ vec(\mathbf{N}_k^T) | \quad (\text{D.5})$$

where  $\circ$  denotes the Hadamard product defined as: if  $\mathbf{A} \in \mathbb{R}^{r \times c}$ ,  $\mathbf{B} \in \mathbb{R}^{r \times c}$ , then  $\mathbf{C} = (\mathbf{A} \circ \mathbf{B}) \in \mathbb{R}^{r \times c}$  with elements  $c_{ij} = a_{ij} \times b_{ij}$ .

While it may be strictly correct to normalise by  $\mathbf{N}_1^T$ , this has an inherent problem in that if the centre is zero then the absolute normalised difference would be infinity.

To assess the *engineering significance* (Montgomery, 1991) of  $\Omega_k$  it is useful to compare it to a threshold value, denoted as  $\beta$ . Therefore,  $\Omega_k \leq \beta$  and  $\Omega_k > \beta$  indicate that  $\Omega_k$  was acceptable (less than or equal to the threshold) and unacceptable (more than the threshold), respectively. Furthermore,  $\Omega_k(\psi)$ ,  $\psi \in \mathbb{R}$  denotes the explicit value. Thus  $\Omega_k(\psi) \leq \beta$  indicates the vicinity of the  $\Omega_k(\psi)$  to the threshold.

An equivalent compact notation can be developed for comparing corresponding cluster correlation matrices. Due to their symmetry the  $vech(\bullet)$  operator can be utilised. Let  $\mathbf{C}_k$  denote the matrix:

$$\mathbf{C}_k = [vech({}_k\mathbf{R}_1), \cdots, vech({}_k\mathbf{R}_f)] \quad \forall i \in \mathbb{N} \wedge i \in [1, nc] \quad (\text{D.6})$$



where  $\mathbf{R}$  is the correlation matrix for the  $k^{th}$  sub-sample for the  $nc^{th}$  cluster and each column of  $\mathbf{C}_k$  denotes the unique elements of the  $i^{th}$  correlation matrix. If  $\mathbf{C}_1$  represents the matrix of correlations for the largest or *reference* population then, analogous to  $\Omega_k$ , the operator  $\Phi_k$  can be defined as:

$$\Phi_k = \max | \text{vec}(\mathbf{C}_1 - \mathbf{C}_k) | \quad (\text{D.7})$$

If we let  $\delta c$  represent a critical threshold for  $\mathbf{R}_k$ , then we can write  $\Phi_k(\omega) \leq \delta c$  to denote an acceptable difference and  $\Phi_k(\omega) > \delta c$  to indicate an unacceptable difference in correlations. We can combine  $\Omega_k$  and  $\Phi_k$  together using Boolean operators to produce composite conditions, *e.g.*  $\Omega_k(\psi) \leq \beta \wedge \Phi_k(\omega) \leq \delta c$  would indicate that both the maximum normalised differences in cluster centres and absolute cluster correlation difference matrix are below their respective thresholds. In contrast,  $\Omega_k(\psi) \leq \beta \wedge \Phi_k(\omega) > \delta c$  would fail the maximum allowable absolute correlation differences only.

The consistent cluster data membership condition can be combined using Boolean operators with the notation previously defined for  $\Omega_k$  and  $\Phi_k$ . For example, the statement:

$$\{ {}_k D \subset {}_1 D_i \}_{i=1}^{nc} \wedge \Omega_k(\psi) \leq \beta \wedge \Phi_k(\omega) \leq \delta c \quad (\text{D.8})$$

defines a valid sub-sample cluster analysis.

### D.3 Cluster Bound Notation used in CR4

After a clustering analysis has been carried out to determine a valid sub-sampled POP that agrees with the reference POP, a cluster bound analysis is conducted. The purpose of this is to discard any clusters from subsequent analyses if the cluster bounds exceed the engineering significance for one or more specific objectives. Let  $\mathbf{X}_k \in \mathbb{R}^{n_k \times nobj}$  denote the matrix of POP solutions for the  $k^{th}$  cluster with individual elements  ${}_k x_{\gamma m}$ , for the  $m^{th}$  objective where  $\gamma$  is the sub-sample size,  $q \in [1, nobj]$  and  $nobj$  is the number of objectives. In order to test if the cluster bounds for a particular objective exceed this engineering significance, statements such as the following can be used:

$${}_k x_{\gamma m} \in [lo_m, up_m] \quad (\text{D.9})$$

where  $lo_m$  and  $up_m$  denote the lower and upper limits on engineering significance as applied to the  $m^{th}$  objective. Such limits from hereon will be referred to as *engineering limits*.

This notation can be developed to explicitly identify conditions at which the engineering limits have been exceeded. To compactly denote the simultaneous application of Equation D.9 for all  $\gamma \in [1, n_k]$ , then:

$${}_k\mathbf{x}_m \in [\mathbf{lo}_m, \mathbf{up}_m] \quad (\text{D.10})$$

Furthermore, if at least one of the  $n$ -engineering limit conditions for the  $m^{th}$  objective is violated, then:

$${}_k\mathbf{x}_m \notin [\mathbf{lo}_m, \mathbf{up}_m] \quad (\text{D.11})$$

Such notation is useful to indicate that some of the sub-sampled POP solutions for a particular objective exceed the corresponding engineering limits, but the precise details are not required. This principle can be extended further to cater for all objectives. If the matrix,  $\mathbf{L} = [\mathbf{l}_1, \mathbf{lo}_2, \dots, \mathbf{l}_{nobj}] \in \mathbb{R}^{n_k \times nobj}$  and  $\mathbf{U} = [\mathbf{u}_1, \mathbf{up}_2, \dots, \mathbf{u}_{nobj}] \in \mathbb{R}^{n_k \times nobj}$ , then to denote that, for all of the sub-sampled POP solutions and all objectives, all the engineering limits are simultaneously satisfied:

$$\mathbf{X}_k \in [\mathbf{L}, \mathbf{U}] \quad (\text{D.12})$$

and if, for at least one objective, at least one engineering limit is violated:

$$\mathbf{X}_k \notin [\mathbf{L}, \mathbf{U}] \quad (\text{D.13})$$

## D.4 An Example Application of the Clustering Notation

The notation is sufficiently flexible to represent a sequence or tree of sub-sample cluster analyses. An example is given below from the six objective diesel problem described in Chapter 5, where the notation for each Cluster verification Rule is listed with a brief description alongside.

1. **CR1.**  $V_1^6(4000, 0.001, 5, 1500, 0.1) = 2$  - the reference clustering analysis in six objectives, of a population of 4000, a learning rate of 0.001, five initial clusters,

maximum iterations of 1500 and a convergence tolerance of 0.1 resulted in two clusters.

2. **CR2(i).**  $V_2^6(2000, 0.001, 5, 1500, 0.1) = 2$  - a clustering analysis on a sub-sampled population of 2000 with all other parameters unchanged resulted in two clusters.
3. **CR3(i).**  $\{ {}_2D_i \subset {}_1D_i \}_{i=1}^2 \wedge \Omega_2(0.022) \leq 0.05 \wedge \Phi_2(0.04) \leq 0.1$  - the sub-sampled POP satisfies the consistent cluster data membership condition AND the maximum absolute normalised difference in centres is less than the threshold of 0.05 AND the difference in absolute cluster correlation matrices is below the threshold of 0.1.
4. **CR2(ii).**  $V_3^6(1000, 0.001, 5, 1500, 0.1) = 2$  - a clustering analysis on a sub-sampled POP of 1000 with all other parameters unchanged resulted in two clusters.
5. **CR3(ii).**  $\{ {}_3D_i \subset {}_1D_i \}_{i=1}^2 \wedge \Omega_3(0.028) \leq 0.05 \wedge \Phi_3(0.09) \leq 0.1$  - the sub-sampled POP satisfies the consistent cluster data membership condition AND the maximum absolute normalised difference in centres is less than the threshold of 0.05 AND the difference in absolute cluster correlation matrices is below the threshold of 0.1.
6. **CR2(iii).**  $V_4^6(500, 0.001, 5, 1500, 0.1) = 2$  - a clustering analysis on a sub-sampled POP of 500 with all other parameters unchanged resulted in two clusters.
7. **CR3(iii).**  $\{ {}_4D_i \subset {}_1D_i \}_{i=1}^2 \wedge \Omega_4(0.065) > 0.05 \wedge \Phi_4(0.1) \leq 0.1$  - the sub-sampled POP satisfies the consistent cluster data membership condition AND the maximum absolute normalised difference in centres is *more* than the threshold of 0.05 AND the difference in absolute cluster correlation matrices is below the threshold of 0.1.

The sub-sampling process is terminated at  $p = 500$  as the maximum absolute normalised difference in centres exceeds the threshold. In otherwords, the POP of 1000 is the smallest sub-sampled POP that provides acceptable agreement with the reference POP.



## Appendix E

# Mathematical Notation for Objective Reduction Rules

A compact notation has been developed to represent the application of Objective reduction Rules OR3a), OR3b) and OR3c) as defined in Section 4.5. As with the clustering notation, this notation avoids the need for verbose and repetitive descriptions of the objective reduction process and also, can be used to specify software to automate the analysis.

### E.1 The Included and Excluded Objective Sets

Let  $\zeta = \{i\}_{i=1}^d, \forall d \in \mathbb{N}$  denote the set of objective indices. It is assumed that each objective is permanently mapped onto one corresponding index variable, *e.g.*

$\text{NOx} \rightarrow 1, \text{Parts} \rightarrow 2, \dots$  Similarly, let  $I$  and  $E$  denote the set of indices for those objectives *included* in and *excluded* from future analyses.  $I$  and  $E$  will be referred to as the *included* and *excluded* index sets respectively. Clearly,  $I \subseteq \zeta, E \subset \zeta, I \cap E = \phi$  and  $I \cup E = \zeta$ .

Therefore, statements such as  $1 \in I$  or  $\{1, 5\} \subset E$  can be written, meaning that *objective 1 is a member of the included set* or *objectives 1 and 5 are a subset of the excluded set*. Application of the objective reduction rules OR3a)-c) should result in an assignment of a particular index to either  $I$  or  $E$ , where we are usually only interested in the retained objectives, *i.e.* those in  $I$ .

In order for the notation to be compact and precise, it is necessary to specify the

information required by the objective reduction rules. Therefore, the following subsections cover the sign and magnitude of the PC eigenvector coefficients, objective priorities, the comparison operators and any applicable threshold values.

## E.2 The Sign and Magnitude of the PC Eigenvector Coefficients

With respect to the PC eigenvector coefficients, for the  $p^{th}$  PC,  ${}_pA_-$  and  ${}_pA_+$  denote the set of eigenvector coefficients that are of negative or positive sign respectively *and* whose magnitude exceeds a critical value,  $\theta_p$ . This critical value corresponds to that resulting from the test of significance defined in objective reduction rule OR3a) in Section 4.5, *i.e.*  $m^{-0.5}$  for the  $m^{th}$  objective. The membership of, for example,  ${}_pA_-$  can then be defined as  ${}_pA_-(1, 2, \dots)$ . The threshold,  $\theta$  can be naturally incorporated into statements such as:  ${}_pA_-(1, 2, \dots) \geq \theta_p$  or  ${}_pA_+(1, 2, \dots) \geq \theta_p$ . For example, for *nobj* objectives,  $nobj = 6 \Rightarrow \theta_6 = 0.408$  and if the PC eigenvector is  $\mathbf{v}^p = [-0.7, 0.01, 0.245, -0.1, 0.436, 0.5]^T$ , then  ${}_pA_-(1) \geq 0.408 \Rightarrow |v_1^p| \geq 0.408$ . Likewise,  ${}_pA_-(5, 6) \geq 0.408 \Rightarrow |v_5^p| \geq 0.408 \wedge |v_6^p| \geq 0.408$ , which can be summarised as:

$${}_pA_-(1) \geq 0.408 \wedge {}_pA_+(5, 6) \geq 0.408 \quad (\text{E.1})$$

or, if the threshold is known,

$${}_pA_-(1) \wedge {}_pA_+(5, 6) \quad (\text{E.2})$$

denotes all coefficients of the  $p^{th}$  eigenvector exceed  $\theta_p$ .

## E.3 The Relative Magnitude of the PC Eigenvector Coefficients

Selection of objectives to retain may depend in part on the relative difference in the magnitude of eigenvector coefficients of the same sign. For example, if  $\theta_p = 0.316$  and  ${}_pA_+(1, 3, 5)$  (the magnitude of objectives 1, 3 and 5 exceed  $\theta_p$  and all are of positive sign), one method of discarding objectives is to set aside those which are relatively small compared to those objectives retained. Furthermore, if, for example,  $|v_1^p| = 0.32$ ,  $|v_3^p| = 0.48$  and  $|v_5^p| = 0.5$ , then objective 1 may be discarded on the basis that its

magnitude is noticeably smaller than that of objectives 3 and 5 - smaller by at least 0.1, say. This simple heuristic rule has been applied as part of the objective reduction process followed in the Gasoline Engine Cold Start Calibration Optimisation case study.

Such tests can be written directly in terms of the eigenvector coefficients, *e.g.*

$$|v_3^p| - |v_1^p| > 0.1 \quad (\text{E.3})$$

$$|v_5^p| - |v_1^p| > 0.1 \quad (\text{E.4})$$

$$|v_5^p| - |v_3^p| < 0.1 \quad (\text{E.5})$$

However, the number of combinations soon becomes cumbersome and more importantly, does not directly answer the fundamental question of; '*Which objectives should be retained?*' Consequently, the following notation has been introduced:

$${}_p\Delta_+(i, j, \dots) \geq \delta r, \quad \forall \{i, j, \dots\} \in {}_pA_+ \quad (\text{E.6})$$

$${}_p\Delta_-(r, s, \dots) \geq \delta r, \quad \forall \{r, s, \dots\} \in {}_pA_- \quad (\text{E.7})$$

Applying this to the previous example, we can write  ${}_p\Delta_+(3, 5) \geq 0.1$  to denote that objectives 3 and 5 have been retained for further comparison. If the value of  $\delta$  has been defined, then the notation can be shortened to:

$${}_p\Delta_+(i, j, \dots), \quad \forall \{i, j, \dots\} \in {}_pA_+ \quad (\text{E.8})$$

$${}_p\Delta_-(r, s, \dots), \quad \forall \{r, s, \dots\} \in {}_pA_- \quad (\text{E.9})$$

This notation can be combined with that from Section E.2 on a Boolean basis. For example, either:

$${}_pA_+(1, 3, 5) \wedge {}_p\Delta_+(3, 5) \geq 0.1 \quad (\text{E.10})$$

or,

$${}_pA_+(1, 3, 5) \wedge {}_p\Delta_+(3, 5) \quad (\text{E.11})$$

Equation E.11 implies that, for the  $p^{\text{th}}$  PC, objective 1 has been assigned to E, while objectives 3 and 5 are retained for further comparison.

## E.4 Denoting Objective Priorities

The decision to retain objectives may be strongly influenced by priority. Let  $\mathbf{R} \in \mathbb{R}^{nobj}$  denote the vector of objective priorities and  $r_i$  the  $i^{th}$  element of  $\mathbf{R}$ . Then the statement:

$$r_i \geq r_j, \quad i \neq j \quad (\text{E.12})$$

indicates that the priority assigned to objective  $i$  is equal to or greater than objective  $j$ . Similarly, the statement:

$$r_i \wedge r_j \geq r_m \wedge r_n \quad (\text{E.13})$$

implies that the priorities for objectives  $i$  and  $j$  are either equal to or greater than those for objectives  $m$  and  $n$ .

## E.5 Composite Statements

The notations defined in Sections E.2 and E.4 can be compactly combined to define the conditions under which individual objectives are assigned to  $I$ . For example:

$${}_pA_+(1, 3, 5) \wedge {}_p\Delta_+(3, 5) \wedge r_5 > r_3 \Rightarrow 5 \in I \quad (\text{E.14})$$

suggests that, for the  $p^{th}$  PC, objective 5 has been assigned to I on the following basis:

- The eigenvector coefficients for objectives 1, 3 and 5 for the  $p^{th}$  PC all have a positive sign.
- That  $|v_3^p|$  and  $|v_5^p|$  are sufficiently greater in magnitude than  $|v_1^p|$  so that objective 1 can be discarded.
- The priority assigned to objective 5 is greater than that for objective 3.

Likewise:

$${}_pA_+(1, 3, 5) \wedge {}_p\Delta_+(5) \Rightarrow 5 \in I \quad (\text{E.15})$$

$${}_pA_-(2, 6) \wedge {}_p\Delta_-(2) \Rightarrow 2 \in I \quad (\text{E.16})$$

indicates that the eigenvector coefficients, whose value exceeds  $\theta_p$ , fall into two signed groups, one positive and the other negative. From each of these groups objectives 5



and 2 have been assigned to  $I$  based only on their relative magnitude, as no accompanying priority is specified.

Examples such as that described in Equation E.14 apply in cases where all eigenvector coefficients exceeding  $\theta_p$  have the same sign. This indicates harmony. Conversely, the example described in Equation E.15 comprises eigenvector coefficients with different signs indicating conflict.

The usefulness of the notation will now be demonstrated for a variety of different scenarios as follows:

- ${}_pA_-(i) \Rightarrow i \in I$  - the  $k^{th}$  PC contains only one significant eigenvector coefficient and so magnitude or priority comparisons are not necessary.
- ${}_pA_+(i, j) \wedge r_j > r_i \Rightarrow j \in I$  - objective  $j$  has been assigned to  $I$ , *i.e.* retained, based on priority only.
- ${}_pA_+(i, j) \Rightarrow i \in I$  - indicates that  $||v_i^p| - |v_j^p|| < \delta r$  and  $r_i = r_j$  and objective  $i$  has been assigned to  $I$  as it is larger in magnitude than  $j$ . For brevity, only those tests relevant to the retention or otherwise of objectives are written and so, for example, this is why the priority test was omitted. In this example, if objective  $i$  was equal in magnitude to  $j$ , then the decision maker should still select one objective only, as having the same sign, both objectives are in harmony and one can be discarded.

Up until now, conditions relating to  ${}_pA_-$  and  ${}_pA_+$  have been written on separate lines, but it is possible to combine these on a single line without loss of quality. For example,

$${}_pA_-(i) \wedge {}_pA_+(j) \Rightarrow \{i, j\} \subseteq I \quad (E.17)$$

or,

$${}_pA_-(i) \wedge ({}_pA_+(j, m) \wedge r_j > r_m) \Rightarrow \{i, j\} \subseteq I \quad (E.18)$$

which suggests that objective  $m$  has been discarded due to a lower priority. Again, the absence of a  ${}_p\Delta_+(j)$  statement implies  $||v_i^p| - |v_m^p|| < \delta r$ .

Finally, consider:

$$\begin{aligned} &({}_pA_-(i, j, m) \wedge {}_p\Delta_-(j, m) \wedge r_m > r_j) \wedge ({}_pA_+(r, s, t, u) \wedge \\ &\wedge {}_p\Delta_+(r, s, t) \wedge r_r > r_s > r_t) \Rightarrow \{m, r\} \subseteq I \end{aligned} \quad (E.19)$$

Objective  $m$  is retained from  ${}_pA_-$  based on priority compared to objective  $j$  with both  $j$  and  $m$  larger in magnitude than  $i$ . Likewise for  ${}_pA_+$ , objective  $u$  is the first objective discarded (based on magnitude), then  $s$  and  $t$  based on priority, leaving objective  $r$  (along with  $m$ ) to be retained.

## Appendix F

# Detailed Results from Objective Reduction Process - 1st Stage

This Appendix provides details of the Stage 1 clustering and objective reduction processes summarised in Section 6.6 and depicted in the flowchart in Figure 6.6.

### F.1 Clustering and Verification

#### F.1.1 Establish Reference Clusters

Applying Cluster verification Rule CR1, robust clustering was repeated on the 18,552 population from various settings of learning rate,  $lr$ , and initial number of clusters,  $cs$ . Learning rate was varied as preliminary testing on this problem showed some sensitivity to this parameter. The settings of these parameters was defined using a Stratified Latin Hypercube (SLH) array available in the Matlab® v7.7 (R2008b) Model-Based Calibration Toolbox™. This was chosen as it was readily available in Matlab® and provided an experimental array based on a mixture of integer ( $cs$ ) and continuous ( $lr$ ) inputs. The learning rate parameter was added as a variable in the batch of clustering runs as preliminary testing on this ten-objective problem suggested there may be some dependency of the results on this parameter. The number of clustering runs in this array was 37, based on the number of PC processors available to be run in parallel as batch job (*i.e.* one run per processor) using the Matlab® Parallel Computing Toolbox. The parameters of the k\*-Means clustering algorithm were set as follows:

- Learning rate,  $lr$ : 1e-4 to 1e-6.
- Initial number of clusters,  $cs$ : 5 to 7.
- Convergence tolerance: 0.1.
- Window: 50.
- Maximum iterations: 5000.
- Proportion of data retained for robust clustering (using FAST-MCD): 95%.

The whole batch job of 37 runs took approximately seven hours using Intel® Core™ 2 Quad Q9450 2.66GHz processors with 3.25GB RAM on each PC.

The reference clustering analysis in ten objectives, of a Pareto-optimal population (POP) of 18,552 (the *reference POP*), SLH array of learning rates and initial clusters, maximum iterations of 5000 and a convergence tolerance of 0.1 resulted in a total of seven runs with converged clusters, where the most frequently occurring solution had three clusters. From Section D.2.3, convergence was defined as  $\Omega_1 \leq 0.05 \wedge \Phi_1 \leq 0.1$  - the maximum absolute normalised difference in centres is less than the threshold of 0.05 AND the difference in absolute cluster correlation matrices is below the threshold of 0.1. All the other runs did not converge. Only generating seven converged runs out of a total of 113 runs (three batches of runs were carried out) was disappointing and suggested that there may be issues with the k\*-Means parameter settings or the algorithm itself. These potential problems were not explored further at this stage since some useful results had been generated and consequently, these issues were left for future research. Of the four-cluster solutions, that with the best convergence was selected, *i.e.* run 18 and is referred to from hereon as the *reference solution*.

### F.1.2 Establish Sub-sampled POP Size

Subsequently, the reference solution was randomly sub-sampled per cluster using the Matlab® routine *randperm* to generate smaller POPs of 10,000, 5000, 2000 and 1000. Robust clustering was run on all of the sub-sampled POPs to test for agreement with the reference solution clusters. The results from the application of the Clustering verification Rules are summarised below using the notation (defined in Appendix D) together with an explanatory description:

1. **CR1.**  $V_1^{10}(18552, lr, cs, 5000, 0.1) = 4$  - the reference clustering analysis in ten objectives, of the reference POP of 18,552, SLH array of learning rates,  $lr$ , and initial number of clusters,  $cs$ , maximum iterations of 5000 and a convergence tolerance of 0.1 resulted in three runs of four converged clusters, which were very similar to each other in terms of centres and correlation matrices.
2. **CR2(i).**  $V_2^{10}(10000, lr, cs, 5000, 0.1) = [4, 4, 4]$  - a clustering analysis on a sub-sampled population of 10,000 with the SLH array of learning rates,  $lr$  and initial number of clusters,  $cs$  and all other parameters unchanged resulted in three different solutions of four clusters. A run with the best convergence was selected from those in the first (most frequently occurring) solution.
3. **CR3(i).**  $\{ {}_2D_i \subset {}_1D_i \}_{i=1}^4 \wedge \Omega_2(0.04) \leq 0.05 \wedge \Phi_2(0.06) \leq 0.1$  - the sub-sampled POP of 10,000 satisfies the consistent cluster data membership condition AND the maximum absolute normalised difference in centres of 0.04 is less than the threshold of 0.05 AND the difference in absolute cluster correlation matrices of 0.06 is below the threshold of 0.1.
4. **CR2(ii).**  $V_3^{10}(5000, lr, cs, 5000, 0.1) = [4, 4]$  - a clustering analysis on a sub-sampled POP of 5000 with the SLH array of  $lr$  and  $cs$  and all other parameters unchanged resulted in two different solutions of four clusters. A run with the best convergence was selected from those in the first solution.
5. **CR3(ii).**  $\{ {}_3D_i \subset {}_1D_i \}_{i=1}^4 \wedge \Omega_3(0.06) > 0.05 \wedge \Phi_3(0.07) \leq 0.1$  - the sub-sampled POP of 5000 does satisfy the consistent cluster data membership condition AND the difference in absolute cluster correlation matrices is below the threshold of 0.1, but the maximum absolute normalised difference in centres is *not* less than the threshold of 0.05.
6. **CR2(iii).**  $V_4^{10}(2000, lr, cs, 5000, 0.1) = [4, 4]$  - a clustering analysis on a sub-sampled POP of 2000 with the SLH array of  $lr$  and  $cs$  and all other parameters unchanged resulted in two different solutions of four clusters. A run with the best convergence was selected from those in the first solution.
7. **CR3(iii).**  $\{ {}_4D_i \subset {}_1D_i \}_{i=1}^4 \wedge \Omega_4(0.09) > 0.05 \wedge \Phi_4(0.07) \leq 0.1$  - the sub-sampled POP of 2000 does satisfy the consistent cluster data membership condition

AND the difference in absolute cluster correlation matrices is below the threshold of 0.1, but the maximum absolute normalised difference in centres is *not* less than the threshold of 0.05.

8. **CR2(iv).**  $V_5^{10}(1000, lr, cs, 5000, 0.1) = [4, 4, 4]$  - a clustering analysis on a sub-sampled POP of 1000 with the SLH array of  $lr$  and  $cs$  and all other parameters unchanged resulted in three different solutions of four clusters. A run with the best convergence was selected from those in the first solution.
9. **CR3(iv).**  $\{ {}_5D_i \subset {}_1D_i \}_{i=1}^4 \wedge \Omega_5(0.11) > 0.05 \wedge \Phi_5(0.1) = 0.1$  - the sub-sampled POP satisfies the consistent cluster data membership condition, but the maximum absolute normalised difference in centres is *more* than the threshold of 0.05 AND the difference in absolute cluster correlation matrices is *not* below the threshold of 0.1.

The sub-sampling process could have been terminated at  $p = 5000$  as the maximum absolute normalised difference in centres exceeds the threshold. However, it was decided to continue with the sub-sampling process to check that the trend of increasing disagreement (between the reference and sub-sample clusters) continued with smaller sub-samples.

Tables F.1 and F.2 show the selected clusters for the reference POP compared to those for each of the sub-sampled POPs for Clusters 1/2 and Clusters 3/4 respectively. Table F.3 summarises the differences in cluster centres and correlation matrices for each sub-sample compared to the reference POP as defined by the previously used notation.

In conclusion, it was decided that the POP of 10,000 was the smallest sub-sampled POP that provided acceptable agreement with the reference POP. Considerations in making this decision were:

- The computational expense (*i.e.* approximately seven hours) involved in optimising or clustering a POP of 10,000 with a parallel computing network was not considered prohibitive compared to smaller POPs.
- As stated in Section 2.2, the population size necessary to approximate the Pareto-optimal front rises exponentially with the number of objectives. As a

result, it was decided to be prudent and keep the POP size larger rather than smaller.

Consequently, the sub-sampled POP of 10,000 and associated selected clusters were chosen to continue with the objective reduction process.

Table F.1: Comparison of Clusters 1 and 2 resulting from the POP of 18,552 with those of randomly sub-sampled POPs.

Cluster Property	POP Size	Cluster 1										Cluster 2									
Centres	18,552	-106.9	-45.4	0.8	0.4	2.1	1697.9	0.5	0.1	23.8	5.8	-83.9	-44.0	1.2	0.3	2.2	1593.9	0.2	0.2	7.7	15.0
	10000	-106.9	-45.6	0.8	0.4	2.1	1697.1	0.5	0.1	23.4	5.8	-84.3	-44.7	1.2	0.3	2.2	1595.1	0.2	0.2	7.8	14.7
	5000	-106.8	-45.7	0.8	0.4	2.1	1697.2	0.5	0.1	23.2	5.9	-84.5	-45.3	1.2	0.3	2.2	1595.1	0.2	0.2	7.9	14.5
	2000	-107.2	-45.6	0.8	0.4	2.1	1695.5	0.5	0.1	23.9	5.8	-85.0	-45.5	1.2	0.3	2.2	1597.7	0.2	0.2	8.0	13.7
	1000	-106.6	-46.3	0.8	0.4	2.1	1692.2	0.5	0.1	22.9	6.2	-84.1	-43.7	1.2	0.3	2.3	1594.9	0.2	0.2	7.9	14.2
Correlation Matrices	18,552	1										1									
		0.43	1									0.65	1								
		0.82	0.20	1								-0.12	-0.70	1							
		0.30	0.75	-0.04	1							0.61	0.90	-0.68	1						
		0.12	0.40	-0.11	0.57	1						0.58	0.92	-0.75	0.89	1					
		0.44	0.70	0.44	0.59	-0.09	1					-0.48	-0.15	0.14	-0.06	-0.31	1				
		-0.59	-0.72	-0.42	-0.66	-0.36	-0.62	1				0.03	0.25	-0.35	0.37	0.13	0.31	1			
		-0.34	-0.93	-0.11	-0.75	-0.24	-0.76	0.67	1			-0.42	-0.76	0.42	-0.65	-0.63	-0.15	-0.03	1		
		0.20	0.33	-0.03	0.50	0.42	0.22	-0.16	-0.43	1		-0.24	-0.56	0.43	-0.46	-0.40	-0.23	-0.39	0.62	1	
		0.13	-0.45	0.22	-0.58	-0.23	-0.40	0.28	0.41	0.19	1	0.35	0.37	-0.02	0.13	0.22	-0.11	-0.21	-0.46	-0.40	1
	10000	1										1									
		0.43	1									0.64	1								
		0.82	0.19	1								-0.09	-0.69	1							
		0.30	0.75	-0.04	1							0.60	0.90	-0.68	1						
		0.12	0.40	-0.11	0.56	1						0.56	0.92	-0.74	0.88	1					
		0.44	0.70	0.44	0.58	-0.10	1					-0.47	-0.15	0.12	-0.05	-0.30	1				
		-0.58	-0.72	-0.40	-0.66	-0.36	-0.61	1				0.02	0.23	-0.33	0.35	0.11	0.31	1			
		-0.34	-0.93	-0.11	-0.75	-0.24	-0.75	0.67	1			-0.42	-0.77	0.42	-0.65	-0.64	-0.14	-0.01	1		
		0.21	0.34	-0.02	0.49	0.41	0.22	-0.16	-0.43	1		-0.23	-0.55	0.41	-0.44	-0.39	-0.22	-0.38	0.61	1	
		0.14	-0.44	0.23	-0.57	-0.23	-0.39	0.28	0.39	0.21	1	0.36	0.37	-0.01	0.11	0.21	-0.12	-0.22	-0.46	-0.41	1
	5000	1										1									
		0.39	1									0.61	1								
		0.81	0.12	1								-0.04	-0.67	1							
		0.27	0.74	-0.09	1							0.57	0.89	-0.66	1						
		0.13	0.43	-0.12	0.60	1						0.54	0.92	-0.73	0.87	1					
		0.42	0.67	0.42	0.54	-0.08	1					-0.49	-0.14	0.11	-0.05	-0.31	1				
		-0.58	-0.70	-0.39	-0.64	-0.37	-0.59	1				0.00	0.23	-0.34	0.35	0.11	0.29	1			
		-0.28	-0.93	-0.03	-0.75	-0.27	-0.72	0.65	1			-0.37	-0.74	0.37	-0.61	-0.59	-0.17	0.00	1		
		0.17	0.32	-0.07	0.49	0.44	0.17	-0.13	-0.42	1		-0.19	-0.54	0.40	-0.43	-0.37	-0.25	-0.38	0.61	1	
		0.15	-0.43	0.24	-0.55	-0.22	-0.38	0.26	0.40	0.22	1	0.36	0.35	0.03	0.09	0.19	-0.12	-0.22	-0.46	-0.37	1
	2000	1										1									
		0.51	1									0.61	1								
		0.81	0.26	1								0.00	-0.65	1							
		0.35	0.74	-0.01	1							0.54	0.88	-0.64	1						
		0.17	0.41	-0.08	0.58	1						0.54	0.91	-0.69	0.86	1					
		0.51	0.69	0.51	0.54	-0.09	1					-0.54	-0.18	0.08	-0.03	-0.33	1				
		-0.60	-0.74	-0.41	-0.66	-0.36	-0.63	1				-0.06	0.17	-0.33	0.30	0.05	0.31	1			
		-0.41	-0.93	-0.16	-0.76	-0.26	-0.74	0.70	1			-0.33	-0.73	0.38	-0.60	-0.58	-0.16	0.04	1		
		0.22	0.28	-0.04	0.46	0.44	0.12	-0.10	-0.38	1		-0.19	-0.52	0.37	-0.40	-0.35	-0.20	-0.35	0.58	1	
		0.07	-0.44	0.16	-0.53	-0.21	-0.40	0.30	0.41	0.26	1	0.36	0.34	0.05	0.08	0.19	-0.16	-0.26	-0.44	-0.38	1
	1000	1										1									
		0.29	1									0.57	1								
		0.75	-0.02	1								-0.09	-0.72	1							
		0.16	0.75	-0.23	1							0.54	0.89	-0.72	1						
		0.07	0.46	-0.22	0.57	1						0.48	0.91	-0.80	0.89	1					
		0.33	0.67	0.33	0.54	-0.08	1					-0.45	-0.05	0.10	0.01	-0.21	1				
		-0.55	-0.64	-0.33	-0.56	-0.37	-0.52	1				0.19	0.35	-0.31	0.45	0.24	0.24	1			
		-0.19	-0.92	0.08	-0.75	-0.28	-0.73	0.57	1			-0.34	-0.73	0.41	-0.60	-0.59	-0.21	-0.02	1		
		0.08	0.39	-0.25	0.53	0.50	0.15	-0.11	-0.46	1		-0.23	-0.56	0.37	-0.42	-0.38	-0.26	-0.33	0.64	1	
		0.17	-0.46	0.27	-0.59	-0.20	-0.45	0.25	0.44	0.13	1	0.31	0.31	0.04	0.06	0.12	-0.07	-0.20	-0.43	-0.42	1

Table F.2: Comparison of Clusters 3 and 4 resulting from the POP of 18,552 with those of randomly sub-sampled POPs.

Cluster Property	POP Size	Cluster 1										Cluster 2									
Centres	18,552	-79.3	-38.3	0.7	0.6	2.8	1447.1	0.6	0.1	8.8	14.0	-60.5	-75.3	1.6	0.4	1.7	1541.8	0.2	0.0	25.0	47.2
	10000	-79.4	-38.3	0.7	0.6	2.8	1447.8	0.6	0.1	8.7	14.3	-60.6	-75.1	1.6	0.4	1.7	1542.3	0.2	0.0	24.9	46.7
	5000	-79.5	-38.4	0.8	0.6	2.8	1448.0	0.6	0.1	8.8	14.4	-60.4	-75.2	1.6	0.4	1.7	1541.4	0.2	0.0	25.2	47.4
	2000	-79.5	-38.5	0.8	0.6	2.8	1449.9	0.6	0.1	8.8	14.2	-60.4	-75.2	1.6	0.4	1.7	1542.8	0.2	0.0	24.8	47.0
	1000	-79.1	-38.2	0.7	0.6	2.8	1444.2	0.6	0.1	9.0	14.0	-60.4	-75.0	1.6	0.4	1.7	1541.4	0.2	0.0	25.0	46.0
Correlation Matrices	18,552	1										1									
		0.07	1									-0.72	1								
		-0.23	-0.93	1								0.14	-0.58	1							
		0.86	0.48	-0.60	1							-0.19	0.22	0.19	1						
		0.87	0.23	-0.30	0.79	1						-0.59	0.53	-0.41	0.22	1					
		-0.89	-0.31	0.46	-0.86	-0.76	1					-0.63	0.15	0.43	0.03	-0.14	1				
		-0.68	-0.16	0.23	-0.55	-0.86	0.55	1				-0.42	0.34	0.14	0.47	0.18	0.22	1			
		0.14	-0.09	0.14	-0.02	0.29	0.01	-0.21	1			-0.15	0.05	-0.09	0.02	0.24	0.05	-0.16	1		
		0.32	-0.30	0.18	0.13	0.26	-0.23	-0.32	0.10	1		0.60	-0.12	-0.21	-0.33	-0.25	-0.56	-0.40	-0.22	1	
		-0.52	0.00	0.07	-0.36	-0.69	0.41	0.91	0.02	-0.33	1	0.18	-0.26	0.17	-0.12	-0.16	-0.13	0.08	-0.26	0.39	1
	10000	1										1									
		0.07	1									-0.73	1								
		-0.22	-0.93	1								0.16	-0.58	1							
		0.85	0.48	-0.61	1							-0.19	0.24	0.16	1						
		0.86	0.24	-0.30	0.78	1						-0.59	0.54	-0.42	0.21	1					
		-0.89	-0.33	0.47	-0.86	-0.76	1					-0.63	0.16	0.41	0.03	-0.13	1				
		-0.68	-0.15	0.22	-0.54	-0.86	0.55	1				-0.41	0.34	0.14	0.47	0.18	0.22	1			
		0.13	-0.10	0.15	-0.04	0.29	0.03	-0.22	1			-0.17	0.06	-0.12	-0.02	0.26	0.05	-0.17	1		
		0.33	-0.32	0.20	0.13	0.26	-0.24	-0.31	0.12	1		0.60	-0.13	-0.18	-0.32	-0.25	-0.56	-0.39	-0.21	1	
		-0.52	0.02	0.05	-0.35	-0.69	0.40	0.91	0.01	-0.32	1	0.16	-0.26	0.20	-0.12	-0.16	-0.10	0.07	-0.24	0.38	1
	5000	1										1									
		0.13	1									-0.69	1								
		-0.27	-0.94	1								0.09	-0.54	1							
		0.85	0.54	-0.65	1							-0.25	0.27	0.19	1						
		0.87	0.30	-0.36	0.80	1						-0.56	0.47	-0.36	0.21	1					
		-0.89	-0.40	0.53	-0.88	-0.79	1					-0.66	0.17	0.44	0.09	-0.12	1				
		-0.69	-0.20	0.26	-0.57	-0.86	0.58	1				-0.42	0.32	0.18	0.49	0.15	0.25	1			
		0.14	-0.10	0.14	-0.02	0.27	0.02	-0.19	1			-0.16	0.05	-0.10	-0.02	0.18	0.08	-0.15	1		
		0.32	-0.30	0.18	0.12	0.24	-0.22	-0.29	0.13	1		0.60	-0.11	-0.21	-0.38	-0.23	-0.57	-0.41	-0.20	1	
		-0.54	-0.04	0.10	-0.38	-0.69	0.43	0.91	0.04	-0.31	1	0.12	-0.23	0.17	-0.15	-0.12	-0.09	0.05	-0.22	0.38	1
	2000	1										1									
		0.15	1									-0.68	1								
		-0.28	-0.94	1								0.01	-0.49	1							
		0.85	0.55	-0.65	1							-0.31	0.30	0.18	1						
		0.86	0.35	-0.39	0.79	1						-0.51	0.44	-0.34	0.27	1					
		-0.88	-0.43	0.55	-0.88	-0.78	1					-0.67	0.17	0.49	0.09	-0.17	1				
		-0.69	-0.22	0.28	-0.55	-0.85	0.57	1				-0.47	0.41	0.19	0.50	0.12	0.30	1			
		0.14	-0.05	0.09	0.01	0.28	0.03	-0.20	1			-0.15	0.03	-0.06	-0.02	0.19	0.08	-0.15	1		
		0.31	-0.28	0.16	0.11	0.21	-0.23	-0.26	0.10	1		0.62	-0.13	-0.26	-0.43	-0.22	-0.59	-0.42	-0.22	1	
		-0.52	-0.04	0.10	-0.35	-0.68	0.42	0.91	0.05	-0.27	1	0.14	-0.23	0.12	-0.14	-0.11	-0.13	0.03	-0.20	0.40	1
	1000	1										1									
		0.11	1									-0.64	1								
		-0.25	-0.93	1								-0.09	-0.45	1							
		0.87	0.48	-0.61	1							-0.31	0.31	0.20	1						
		0.89	0.29	-0.36	0.82	1						-0.44	0.42	-0.31	0.31	1					
		-0.91	-0.33	0.48	-0.88	-0.82	1					-0.72	0.17	0.51	0.07	-0.17	1				
		-0.75	-0.20	0.29	-0.62	-0.87	0.65	1				-0.48	0.36	0.26	0.49	0.10	0.32	1			
		0.03	-0.12	0.21	-0.11	0.17	0.14	-0.12	1			-0.20	0.09	-0.11	-0.01	0.21	0.09	-0.13	1		
		0.34	-0.35	0.21	0.14	0.25	-0.27	-0.32	0.07	1		0.61	-0.06	-0.30	-0.41	-0.19	-0.57	-0.41	-0.25	1	
		-0.62	-0.05	0.16	-0.46	-0.72	0.54	0.92	0.10	-0.35	1	0.14	-0.21	0.15	-0.15	-0.20	-0.08	0.10	-0.30	0.40	1



Table F.3: Comparison of the selected reference clusters from the reference (18,552) POP with those of randomly sub-sampled POPs, showing the maximum absolute normalised difference in cluster centres and the maximum absolute difference in cluster correlation matrices.

Sub-sample POP size	Cluster	Maximum absolute normalised difference in cluster centres	Maximum absolute difference in cluster correlation matrices
10000	1	0.04	0.06
	2	0.02	0.01
	3	0.03	0.01
	4	0.01	0.03
5000	1	0.06	0.03
	2	0.03	0.03
	3	0.02	0.01
	4	0.06	0.07
2000	1	0.05	0.07
	2	0.09	0.05
	3	0.06	0.03
	4	0.04	0.05
1000	1	0.11	0.03
	2	0.07	0.04
	3	0.04	0.03
	4	0.05	0.10

### F.1.3 Check Cluster Bounds

Using the notation defined in Section D.3 for CR4, the only engineering limit that was specified by the Cold Start Calibration Engineer, was applied to the Peak\_Flare\_Speed Objective, where only solutions in the 1300 to 1500rpm range were of interest. Therefore, Equations D.10 and D.11 were applied as follows:

$$1x_{10000,6} \notin [1300_6, 1500_6] \quad (\text{F.1})$$

$$2x_{10000,6} \in [1300_6, 1500_6] \quad (\text{F.2})$$

$$3x_{10000,6} \in [1300_6, 1500_6] \quad (\text{F.3})$$

$$4x_{10000,6} \in [1300_6, 1500_6] \quad (\text{F.4})$$

In other words, Cluster 1 was discarded as it only contained data in a excessively high range for Objective 6, Peak\_Flare\_Speed and all other clusters were retained as they satisfied the specified engineering limits. The cluster data bounds per objective are

detailed in Table F.4, where Objective 6 in Cluster 1 is highlighted in grey.

Table F.4: Cluster membership and bounds on data per cluster for the selected 10,000 POP; Cluster 1 discarded due to excessively high range of Objective 6, Peak\_Flare\_Speed, as highlighted in grey.

Cluster	Membership	Bounds											
		Upper	-78.9	-35.5	1.4	0.5	2.8	1812.1	1.5	0.4	56.3	20.7	
1	2958	Lower	-120.5	-61.5	0.4	0.2	1.7	1567.7	0.0	0.0	0.0	0.0	
2	2435	Upper	-63.1	-4.1	1.7	0.6	3.1	1761.2	0.6	0.4	30.9	67.3	
		Lower	-110.8	-76.6	0.7	0.2	1.6	1453.8	0.0	0.0	0.0	0.0	
3	2173	Upper	-45.6	-41.4	1.8	0.6	2.4	1670.6	0.9	0.2	37.3	85.9	
		Lower	-96.9	-94.1	1.2	0.2	1.5	1445.4	0.0	0.0	0.1	0.0	
4	2435	Upper	-65.5	-31.2	1.3	0.8	3.5	1583.4	1.8	0.3	31.0	40.8	
		Lower	-94.1	-51.0	0.4	0.4	1.9	1332.7	0.0	0.0	0.0	0.0	

## F.2 PCA and Potential Objective Reduction

Table F.5 summarises the results of applying the objective reduction rules to each cluster. This was carried out for retained PCs corresponding to three different percentages of total variation to understand the effect on objective reduction. Details of how these rules are applied using the notation is provided below. The exception is Cluster 2 for PCs retained to account for 95% of the total variation, which is given as an example in Section 6.6.2. The Cold Start calibration engineer advised objective priorities as shown in Table 6.3 and these were taken into account where necessary when applying the objective reduction rules.

### F.2.1 Retaining PCs to account for 95% of the Total Variation

#### Objective Reduction Rules applied to Cluster 3

1. Applying OR1,  $\lambda_1/\lambda_{10} = 428.5$ , so there was not sufficient evidence of a pure linear dependency.
2. Applying OR2,  $t_7 = 95\%$ , *i.e.* first seven PCs were retained, which accounted for approximately 95% of the cumulative percentage total variation.

Table F.5: Clustering and PCA results for the selected POP. For each cluster, the eigenvalues are listed in Principal Component order (largest first), followed by the corresponding cumulative percentage of total variation. Also, the rotated eigenvectors are listed by objective for the retained PCs with the selected significant eigenvector coefficients shown shaded.

		Cluster 2					Cluster 3							Cluster 4			
PC Eigenvalues ( $\lambda_p$ )		5.2142					3.2296							5.1305			
		1.7960					2.0245							2.5085			
		1.3700					1.3707							1.0173			
		0.6374					1.0689							0.7364			
		0.4895					0.9628							0.3895			
		0.2771					0.5920							0.1311			
		0.0977					0.3789							0.0461			
		0.0649					0.3036							0.0187			
		0.0380					0.0615							0.0119			
	0.0152					0.0075							0.0100				
Cumulative % of Total Variation		52.1					32.3							51.3			
		70.1					52.5							76.4			
		83.8					66.2							86.6			
		90.2					76.9							93.9			
		95.1					86.6							97.8			
		97.8					92.5							99.1			
		98.8					96.3							99.6			
		99.5					99.3							99.8			
		99.8					99.9							99.9			
		100					100							100			
PCs retained for 95% of total variation		PC1	PC2	PC3	PC4	PC5	PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC1	PC2	PC3	PC4
Rotated Eigenvectors ( $\mathbf{v}^p$ )	Obj1	-0.2787	0.2012	0.0401	0.7149	0.1345	-0.5445	0.1954	-0.0276	-0.0521	0.0093	-0.2794	0.0887	-0.4823	0.0947	-0.0718	0.0632
	Obj2	-0.4396	-0.0045	0.0762	0.0069	-0.0009	0.2249	-0.7451	0.0139	-0.1577	0.0356	-0.0264	0.0186	-0.0433	-0.6209	-0.0279	0.0043
	Obj3	0.3149	-0.2304	-0.0567	0.6459	-0.1307	0.3202	0.5510	0.1280	-0.3497	0.0860	-0.0558	0.1038	0.1539	0.5652	-0.0990	0.0084
	Obj4	-0.4734	-0.0909	-0.1260	0.1389	0.1446	-0.0829	-0.0504	-0.0498	-0.9117	-0.0299	0.0324	-0.0468	-0.5192	-0.1346	0.0074	-0.0458
	Obj5	-0.4881	0.1014	-0.0740	-0.1270	-0.1049	-0.0314	0.0199	0.0231	-0.0307	0.0127	0.9201	0.0547	-0.2809	-0.0267	-0.2041	0.3093
	Obj6	0.0333	-0.8487	-0.0322	-0.0124	0.0677	0.7122	0.0500	-0.0257	0.0278	-0.0087	-0.1847	0.0647	0.5304	0.0153	-0.0767	0.0321
	Obj7	0.0196	-0.0179	0.0352	0.0247	0.8753	-0.0553	-0.0873	0.1298	-0.0618	0.0359	-0.0864	-0.8628	0.0180	0.0532	0.0933	-0.6089
	Obj8	0.3965	0.3673	-0.1677	-0.0350	0.2586	0.0108	-0.0023	0.0182	-0.0208	-0.9913	-0.0124	0.0174	0.0272	0.0190	-0.9434	-0.0051
	Obj9	0.0481	0.1358	-0.5910	0.1411	-0.2523	-0.1728	-0.2843	0.4515	-0.0961	0.0692	-0.1563	0.4596	-0.3168	0.5100	0.0870	-0.0363
	Obj10	0.0809	0.1178	0.7671	0.1197	-0.1781	0.0469	0.0956	0.8706	0.0612	-0.0358	0.0660	-0.1251	-0.1035	-0.0614	-0.1747	-0.7246
PCs retained for 90% of total variation		PC1	PC2	PC3	PC4		PC1	PC2	PC3	PC4	PC5	PC6		PC1	PC2	PC3	PC4
Rotated Eigenvectors ( $\mathbf{v}^p$ )	Obj1	-0.5833	0.0818	-0.1353	0.4236		-0.5780	0.1451	-0.0426	0.0216	-0.0010	-0.2341		-0.4823	0.0947	-0.0718	0.0632
	Obj2	-0.3947	0.0021	0.1394	-0.1139		0.2351	-0.6928	-0.0203	-0.2012	0.0499	-0.0590		-0.0433	-0.6209	-0.0279	0.0043
	Obj3	0.0906	0.0751	-0.0439	0.7518		0.1528	0.6178	0.0429	-0.1452	0.0674	-0.0711		0.1539	0.5652	-0.0990	0.0084
	Obj4	-0.4738	-0.2170	0.0250	0.0289		-0.2228	0.0530	-0.2064	-0.7453	-0.0479	0.0443		-0.5192	-0.1346	0.0074	-0.0458
	Obj5	-0.3923	0.0513	-0.0126	-0.2483		0.0103	0.0050	0.0116	0.0286	0.0124	0.9179		-0.2809	-0.0267	-0.2041	0.3093
	Obj6	0.1941	-0.4618	0.4845	0.3481		0.6429	0.1153	-0.0374	0.0801	-0.0054	-0.2403		0.5304	0.0153	-0.0767	0.0321
	Obj7	-0.0927	-0.6674	0.0246	-0.1086		0.1541	-0.0406	0.2326	-0.5941	0.0438	-0.0589		0.0180	0.0532	0.0933	-0.6089
	Obj8	0.2665	-0.0749	-0.4194	-0.1018		0.0144	-0.0219	0.0150	0.0012	-0.9913	-0.0128		0.0272	0.0190	-0.9434	-0.0051
	Obj9	-0.0012	0.0434	-0.5589	0.1910		-0.3149	-0.2946	0.3614	0.1338	0.0671	-0.1637		-0.3168	0.5100	0.0870	-0.0363
	Obj10	0.0131	0.5213	0.4857	0.0009		0.0335	0.1109	0.8757	-0.0665	-0.0362	0.0622		-0.1035	-0.0614	-0.1747	-0.7246
PCs retained for 86% of total variation		PC1	PC2	PC3	PC4		PC1	PC2	PC3	PC4	PC5		PC1	PC2	PC3		
Rotated Eigenvectors ( $\mathbf{v}^p$ )	Obj1	-0.5833	0.0818	-0.1353	0.4236		-0.5637	0.2083	-0.0684	0.0809	0.1637		-0.4215	0.0465	0.0035		
	Obj2	-0.3947	0.0021	0.1394	-0.1139		0.2139	-0.6658	-0.0336	-0.1339	0.0710		-0.0378	-0.6156	-0.0180		
	Obj3	0.0906	0.0751	-0.0439	0.7518		0.1727	0.5941	0.0422	-0.1807	0.0975		0.1096	0.5695	-0.1263		
	Obj4	-0.4738	-0.2170	0.0250	0.0289		-0.2198	0.0814	-0.2405	-0.6882	-0.0709		-0.3785	-0.1951	0.1020		
	Obj5	-0.3923	0.0513	-0.0126	-0.2483		-0.0240	-0.1799	0.1043	-0.2275	-0.6205		-0.4229	-0.0216	-0.1907		
	Obj6	0.1941	-0.4618	0.4845	0.3481		0.6546	0.1272	-0.0506	0.1220	0.1595		0.3919	0.0742	-0.1694		
	Obj7	-0.0927	-0.6674	0.0246	-0.1086		0.1533	-0.0126	0.2024	-0.5386	0.0631		0.3953	-0.0278	0.1651		
	Obj8	0.2665	-0.0749	-0.4194	-0.1018		0.0426	0.1701	-0.0872	0.2745	-0.7126		-0.0290	-0.0180	-0.9259		
	Obj9	-0.0012	0.0434	-0.5589	0.1910		-0.3208	-0.2425	0.3489	0.1810	0.1604		-0.2215	0.4653	0.1416		
	Obj10	0.0131	0.5213	0.4857	0.0009		0.0331	0.1311	0.8665	-0.0660	-0.0759		0.3564	-0.1811	-0.0609		

3. Applying OR3a),  $\theta_{10} = 0.3162$  - the eigenvectors for the seven retained PCs were rotated and the threshold for the test of significance was  $10^{-0.5} = 0.3162$ .
4. Applying OR3c) to PC1,  ${}_1A_-(1) \wedge {}_1A_+(6) \Rightarrow \{1, 6\} \subseteq I$  -  $v_1^1$  and  $v_6^1$  were the only two significant coefficients and so both Objectives 1 (neg\_25\_INT) and 6 (Peak\_Flare\_Speed) were retained.
5. Applying OR3c) to PC2,  ${}_2A_-(2) \wedge {}_2A_+(3) \Rightarrow \{2, 3\} \subseteq I$  -  $v_2^2$  and  $v_3^2$  were the only two significant coefficients and so both Objectives 2 (neg\_612\_INT) and 3 (cyc\_25\_RMS) were retained.
6. Applying OR3b) to PC3,  ${}_3A_+(9, 10) \wedge {}_3\Delta_+(10) \Rightarrow \{10\} \subseteq I$  - the only significant coefficients were  $v_9^3$  and  $v_{10}^3$ , but  $v_{10}^3$  was much larger in magnitude and so Objective 10 (sf\_cyc\_612\_INT\_abs) was retained.
7. Applying OR3b) to PC4,  ${}_4A_-(3, 4) \wedge {}_4\Delta_-(4) \Rightarrow \{4\} \subseteq I$  - the only significant coefficients were  $v_3^4$  and  $v_4^4$ , but  $v_4^4$  was much larger in magnitude and so Objective 4 (cyc\_612\_RMS) was retained.
8. Applying OR3b) to PC5,  ${}_5A_-(8) \Rightarrow \{8\} \subseteq I$  - the only significant coefficient was  $v_8^5$  and so Objective 8 (sf\_cyc\_612\_RMS\_abs) was retained.
9. Applying OR3b) to PC6,  ${}_6A_+(5) \Rightarrow \{5\} \subseteq I$  - the only significant coefficient was  $v_5^6$  and so Objective 5 (F\_model) was retained.
10. Applying OR3c) to PC7,  ${}_7A_-(7) \wedge {}_7A_+(9) \Rightarrow \{7, 9\} \subseteq I$  -  $v_7^7$  and  $v_9^7$  were the only two significant coefficients and so both Objectives 7 (sf\_abs\_cyc\_25\_RMS) and 9 (sf\_cyc\_25\_INT\_abs) were retained.

In summary, all ten objectives were retained in Cluster 3, *i.e.* no objective reduction was achieved.

### Objective Reduction Rules applied to Cluster 4

1. Applying OR1,  $\lambda_1/\lambda_{10} = 512.2$ , so there was insufficient evidence of a pure linear dependency.
2. Applying OR2,  $t_4 = 95\%$ , *i.e.* first four PCs were retained, which accounted for approximately 95% of the cumulative percentage total variation.

3. Applying OR3a),  $\theta_{10} = 0.3162$  - the eigenvectors for the four retained PCs were rotated and the threshold for the test of significance was  $10^{-0.5} = 0.3162$ .
4. Applying OR3c) to PC1,  ${}_1A_-(1, 4, 9) \wedge {}_1A_+(6) \wedge {}_1\Delta_-(1, 4) \wedge (r_1 > r_4) \Rightarrow \{1, 6\} \subseteq I$  - the coefficients,  $v_1^1$ ,  $v_4^1$ ,  $v_6^1$  and  $v_9^1$  were significant.  $v_1^1$ ,  $v_4^1$  and  $v_9^1$  have the same sign, but although  $v_1^1$  was slightly smaller in magnitude than  $v_4^1$ , Objective 1 (neg\_25\_INT) had the highest priority and so was retained.  $v_6^1$  was the only positive significant coefficient and so Objective 6 (Peak\_Flare\_Speed) was also retained.
5. Applying OR3c) to PC2,  ${}_2A_-(2) \wedge {}_2A_+(3, 9) \wedge (r_9 > r_3) \Rightarrow \{2, 9\} \subseteq I$  - the coefficients,  $v_2^2$ ,  $v_3^2$  and  $v_9^2$  were significant.  $v_3^2$  and  $v_9^2$  had the same sign, and although  $v_3^2$  was slightly larger in magnitude than  $v_9^2$ ,  $v_9^2$  had higher priority and so Objective 9 (sf\_cyc\_25\_INT\_abs) was retained along with Objective 2 (neg\_612\_INT).
6. Applying OR3b) to PC3,  ${}_3A_-(8) \Rightarrow \{8\} \subseteq I$  - the only significant coefficient was  $v_8^3$  and so Objective 8 (sf\_cyc\_612\_RMS\_abs) was retained.
7. Applying OR3c) to PC4,  ${}_4A_-(7, 10) \wedge {}_4A_+(5) \wedge {}_4\Delta_-(10) \Rightarrow \{5, 10\} \subseteq I$  - the coefficients,  $v_7^4$  and  $v_{10}^4$  were significant, but  $v_5^4$  was considered sufficiently close in magnitude to the threshold of 0.3162, that it too was considered significant.  $v_7^4$  and  $v_{10}^4$  have the same sign, but  $v_{10}^4$  was larger in magnitude and has higher priority than  $v_7^4$  and so Objective 10 (sf\_cyc\_612\_INT\_abs) was retained.  $v_5^4$  was the only positive significant coefficient and so Objective 5 (F\_model) was also retained.

In summary, seven objectives were retained in Cluster 4:

- neg\_25\_INT
- Peak\_Flare\_Speed
- sf\_cyc\_25\_INT\_abs
- neg\_612\_INT
- sf\_cyc\_612\_RMS\_abs

- sf\_cyc\_612\_INT\_abs
- F\_model

### F.2.2 Retaining PCs to account for 90% of the Total Variation

#### Objective Reduction Rules applied to Cluster 2

1. Applying OR1 gave the same result as above.
2. Applying OR2,  $t_4 = 90\%$ , *i.e.* first four PCs were retained, which accounted for approximately 90% of the cumulative percentage total variation.
3. Applying OR3a),  $\theta_{10} = 0.3162$ .
4. Applying OR3b) to PC1,  ${}_1A_-(1, 2, 4, 5) \wedge {}_1\Delta_-(1) \Rightarrow \{1\} \subseteq I$ .
5. Applying OR3c) to PC2,  ${}_2A_-(6, 7) \wedge {}_2A_+(10) \wedge {}_2\Delta_-(7) \Rightarrow \{7, 10\} \subseteq I$ .
6. Applying OR3c) to PC3,  
 ${}_3A_-(8, 9) \wedge {}_3A_+(6, 10) \wedge {}_3\Delta_-(9) \wedge (r_6 > r_{10}) \Rightarrow \{6, 9\} \subseteq I$ .
7. Applying OR3b) to PC4,  ${}_4A_+(1, 3, 6) \wedge {}_4\Delta_+(3) \Rightarrow \{3\} \subseteq I$ .

In summary, six objectives were retained in Cluster 2:

- neg\_25\_INT
- sf\_cyc\_25\_RMS\_abs
- sf\_cyc\_612\_INT\_abs
- Peak\_Flare\_Speed
- sf\_cyc\_25\_INT\_abs
- cyc\_25\_RMS

#### Objective Reduction Rules applied to Cluster 3

1. Applying OR1, gave the same result as before.
2. Applying OR2,  $t_6 = 90\%$ , *i.e.* the first six PCs were retained, which accounted for approximately 95% of the cumulative percentage total variation.

3. Applying OR3a),  $\theta_{10} = 0.3162$ .
4. Applying OR3c) to PC1,  ${}_1A_-(1) \wedge {}_1A_+(6) \Rightarrow \{1, 6\} \subseteq I$ .
5. Applying OR3c) to PC2,  ${}_2A_-(2) \wedge {}_2A_+(3) \Rightarrow \{2, 3\} \subseteq I$ .
6. Applying OR3b) to PC3,  ${}_3A_+(9, 10) \wedge {}_3\Delta_+(10) \Rightarrow \{10\} \subseteq I$ .
7. Applying OR3b) to PC4,  ${}_4A_-(4, 7) \wedge {}_4\Delta_+(4) \Rightarrow \{4\} \subseteq I$ .
8. Applying OR3b) to PC5,  ${}_5A_-(8) \Rightarrow \{8\} \subseteq I$ .
9. Applying OR3b) to PC6,  ${}_5A_+(5) \Rightarrow \{5\} \subseteq I$ .

In summary, eight objectives were retained in Cluster 3:

- neg\_25\_INT
- Peak\_Flare\_Speed
- neg\_612\_INT
- cyc\_25\_RMS
- sf\_cyc\_612\_INT\_abs
- cyc\_612\_RMS
- sf\_cyc\_612\_RMS\_abs
- F\_model

#### Objective Reduction Rules applied to Cluster 4

The number of PCs retained in Cluster 4 to account for 90% of the variation was the same as for 95% variation. Therefore, seven objectives were retained in Cluster 4:

- neg\_25\_INT
- Peak\_Flare\_Speed
- neg\_612\_INT
- cyc\_25\_RMS

- sf\_cyc\_612\_RMS\_abs
- sf\_cyc\_612\_INT\_abs
- F\_model

### F.2.3 Retaining PCs to account for 86% of the Total Variation

#### Objective Reduction Rules applied to Cluster 2

The number of PCs retained in Cluster 2 to account for 86% of the variation was the same as for 90% variation. Therefore, six objectives were retained in Cluster 2:

- neg\_25\_INT
- sf\_cyc\_25\_RMS\_abs
- sf\_cyc\_612\_INT\_abs
- Peak\_Flare\_Speed
- sf\_cyc\_25\_INT\_abs
- cyc\_25\_RMS

#### Objective Reduction Rules applied to Cluster 3

1. Applying OR1, gave the same result as before.
2. Applying OR2,  $t_5 = 86\%$ , *i.e.* the first five PCs were retained, which accounted for approximately 86% of the cumulative percentage total variation.
3. Applying OR3a),  $\theta_{10} = 0.3162$ .
4. Applying OR3c) to PC1,  ${}_1A_-(1) \wedge {}_1A_+(6) \Rightarrow \{1, 6\} \subseteq I$ .
5. Applying OR3c) to PC2,  ${}_2A_-(2) \wedge {}_2A_+(3) \Rightarrow \{2, 3\} \subseteq I$ .
6. Applying OR3b) to PC3,  ${}_3A_+(9, 10) \wedge {}_3\Delta_+(10) \Rightarrow \{10\} \subseteq I$ .
7. Applying OR3b) to PC4,  ${}_4A_-(4, 7) \wedge {}_4\Delta_-(4) \Rightarrow \{4\} \subseteq I$ .
8. Applying OR3b) to PC5,  ${}_5A_-(5, 8) \wedge (r_5 > r_8) \Rightarrow \{5\} \subseteq I$ .



In summary, seven objectives were retained in Cluster 3:

- neg\_25\_INT
- Peak\_Flare\_Speed
- neg\_612\_INT
- cyc\_25\_RMS
- sf\_cyc\_612\_INT\_abs
- cyc\_612\_RMS
- F\_model

#### Objective Reduction Rules applied to Cluster 4

1. Applying OR1, gave the same result as before.
2. Applying OR2,  $t_3 = 86\%$ , *i.e.* first three PCs were retained, which accounted for approximately 86% of the cumulative percentage total variation.
3. Applying OR3a),  $\theta_{10} = 0.3162$ .
4. Applying OR3c) to PC1,
 
$${}_1A_-(1, 4, 5) \wedge {}_1A_+(6, 7, 10) \wedge (r_5 > r_1 > r_4) \wedge (r_6 > r_{10} > r_7) \Rightarrow \{5, 6\} \subseteq I.$$
5. Applying OR3c) to PC2,  ${}_2A_-(2) \wedge {}_2A_+(3, 9) \wedge {}_2\Delta_+(3) \Rightarrow \{2, 3\} \subseteq I.$
6. Applying OR3b) to PC3,  ${}_3A_-(8) \Rightarrow \{8\} \subseteq I.$

In summary, five objectives were retained in Cluster 4:

- F\_model
- Peak\_Flare\_Speed
- cyc\_25\_RMS
- neg\_612\_INT
- sf\_cyc\_612\_RMS\_abs



## Appendix G

# Detailed Results from Objective Reduction Process - 2nd Stage

This Appendix provides details of the Stage 2 clustering and objective reduction processes summarised in Section 6.7 and depicted in the flowchart in Figure 6.7.

### G.1 Clustering and Verification with Reduced Objectives - 2nd Stage

The final populations resulting from further optimisations with reduced objectives in each of the retained clusters were clustered using the k\*-Means clustering algorithm and initial parameters set as previously in Section 6.6.1. However, two batches of clustering runs were carried out, one with a learning rate of  $1e-4$  to  $1e-7$  and the other with a learning rate of  $1e-3$  to  $1e-5$  in an attempt to find clusters. Neither batch produced any converged clusters.

When examined in more detail, some of the runs that did produce clusters, albeit not converged, had very similar centres. Further investigation revealed in one such run that, in one of the clusters:

- The magnitude of elements within the corresponding covariance matrices were very small. Both the centres and covariance matrices are shown in Table G.1.
- When the objective data in the 10,000 member sub-sampled POP was plotted as a histogram per objective with high-resolution binning, it was obvious that a

Table G.1: Example of a clustering run from the 2nd Stage objective reduction in Cluster 2 showing very similar data between, and very small covariances for, each resulting cluster.

Cluster Property	Cluster	Obj1	Obj3	Obj6	Obj7	Obj9	Obj10
Centres	1	-77.79	1.23	1471.89	0.07	16.91	5.81
	2	-77.80	1.23	1471.91	0.07	16.91	5.80
Covariances	1	0.0027	3.79E-05	-0.0073	1.47E-05	0.0001	-0.0013
		3.79E-05	6.36E-07	-9.37E-05	1.57E-07	2.49E-06	-1.02E-05
		-0.0073	-9.37E-05	0.0327	1.84E-05	0.0004	-0.0008
		1.47E-05	1.57E-07	1.84E-05	4.41E-07	6.24E-06	-3.84E-05
		0.0001	2.49E-06	0.0004	6.24E-06	0.0003	-0.0006
		-0.0013	-1.02E-05	-0.0008	-3.84E-05	-0.0006	0.0035
	2	0.0035	5.22E-05	-0.0106	7.76E-06	-8.21E-05	-0.0006
		5.22E-05	9.94E-07	-0.0001	-1.93E-08	7.59E-07	9.84E-06
		-0.0106	-0.0001	0.0490	4.40E-05	0.0015	-0.0033
		7.76E-06	-1.93E-08	4.40E-05	5.45E-07	8.37E-06	-5.03E-05
		-8.21E-05	7.59E-07	0.0015	8.37E-06	0.0005	-0.0008
		-0.0006	9.84E-06	-0.0033	-5.03E-05	-0.0008	0.0049

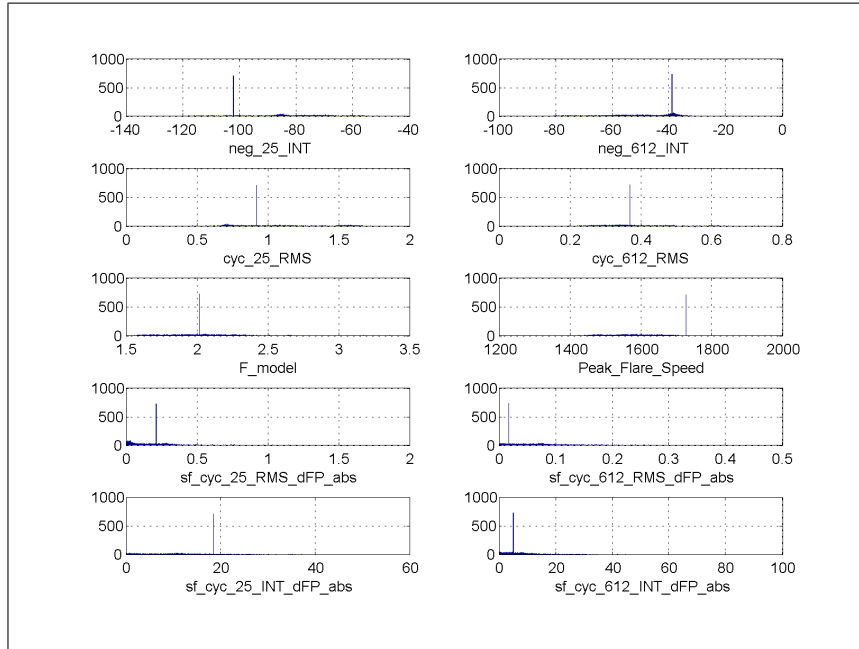


Figure G.1: Histogram plot of each objective in the 10,000 member sub-sampled POP, showing that a large proportion of the data was very similar. The y-axis represents frequency of occurrence.

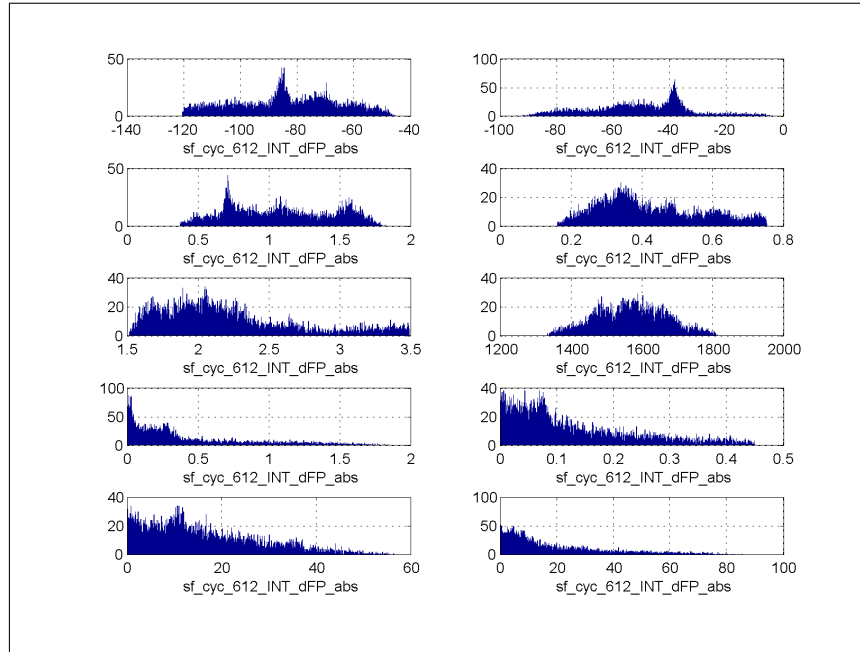


Figure G.2: Histogram plot of each objective in the 10,000 member sub-sampled POP after removing non-unique solutions. The y-axis represents frequency of occurrence.

large number of data was very similar - see Figure G.1. Furthermore, from a powertrain calibration point of view, this similar data was so similar that it was considered as replicate data. Using the Matlab® v7.7 (R2008b) command *unique* to identify unique solutions, non-unique data was removed and the remaining unique 10,000 member sub-sampled POP objective data was re-plotted as histograms, resulting in a much more even distribution of data as shown in Figure G.2.

Using the Matlab® v7.7 (R2008b) command *unique*, individual solutions in a population can be identified. Starting from an initial population size of 10,000, it was found that in each of the final populations within the three retained clusters, there were 2727, 9881, and 5351 unique solutions respectively using the *unique* command with the default Matlab® double precision of 16 decimal places. However, using a user-specified engineering precision of 1 to 3 decimal places to discriminate between solutions revealed only 1043, 742 and 1151 solutions respectively.

As a result, it was decided to reformulate the objective functions so that they were evaluated to this engineering precision in an attempt to generate more unique (to engineering precision) solutions. For example,  $F_{model}$  was reformulated to

$\text{round}(F\_model \times 10^d)/10^d$  where *round* is the Matlab<sup>®</sup> command to round to the nearest integer and  $d$  is the number of decimal places used for engineering precision. Subsequently, the optimisations within each cluster were re-run with the re-formulated objective functions. The resulting final populations produced 2286, 3398 and 2124 unique solutions, which was a significant improvement. The final populations from the optimisations of the re-formulated objectives were then clustered. Again the k\*-Means clustering algorithm was used with initial parameters set as previously in Section 6.6.1 and two batches run at learning rates of: 1e-3 to 1e-5 and 1e-4 to 1e-7 (most runs were 1e-4 to 1e-6). In the retained Clusters 2 and 3, there was no evidence of converged clusters, whereas in Cluster 4, two further clusters were generated. Therefore:

$$\text{For Cluster 2: } V^6(10000, lr, cs, 5000, 0.1) = 1 \quad (\text{G.1})$$

$$\text{For Cluster 3: } V^7(10000, lr, cs, 5000, 0.1) = 1 \quad (\text{G.2})$$

$$\text{For Cluster 4: } V^5(10000, lr, cs, 5000, 0.1) = 2 \quad (\text{G.3})$$

As the computational expense of optimising a population of 10,000 over 10 objectives was considered acceptable with the pMOEA running on more than 30 processors, (2-3 hours), further sub-sampling was not carried out (*i.e.*  $V$  has no suffix). Consequently, Clustering verification Rules, CR2 and CR3 did not apply.

Applying the remaining Clustering verification Rules and the notation as defined in Sections 4.4.3 and Appendix D respectively, to Cluster 4:

- Applying CR1,  $V^5(10000, lr_1, cs, 5000, 0.1) = 2$  and

Table G.2: Cluster membership and bounds on data for each of the two Clusters, 4.1 and 4.2, resulting from the 2nd stage optimisation and clustering within Cluster 4, where n/a is used to denote discarded objectives.

Cluster	Membership	Bounds											
		Upper	n/a	-40.9	1.2	n/a	2.1	1512.9	n/a	0.2	n/a	n/a	n/a
4.1	1140	Lower	n/a	-47.6	0.8	n/a	1.0	1478.1	n/a	0.0	n/a	n/a	n/a
		Upper	n/a	-31.7	1.2	n/a	2.6	1501.3	n/a	0.2	n/a	n/a	n/a
4.2	984	Lower	n/a	-49.1	0.5	n/a	2.0	1427.3	n/a	0.0	n/a	n/a	n/a

$V^5(10000, lr_2, cs, 5000, 0.1) = 2$ , where  $lr_1$ : 1e-4 to 1e-7,  $lr_2$ : 1e-3 to 1e-5 and  $cs$ : 5 to 7. Of the fifteen converged runs generating the same two clusters, the run with the closest agreement between the cluster and data correlation matrices was selected, *i.e.* run 64 using  $lr_1$ .

- Applying CR4, no further engineering limits were applied by the Cold Start calibration engineer. In addition, both clusters had significant membership and so were retained. The bounds on the cluster data are shown in Table G.2.

Table G.3: Stage 2 results from PCA applied to optimisations of reduced objectives within Clusters 2 to 4. Note missing rows in the tables of eigenvectors indicate discarded objectives.

		Cluster 2				Cluster 3				Cluster 4.1				Cluster 4.2				
PC Eigenvalues ( $\lambda_p$ )		3.5134				4.0857				3.8246				3.4529				
		1.9066				1.9230				0.7698				1.2349				
		0.4318				0.8830				0.2613				0.2892				
		0.1235				0.0668				0.1411				0.0203				
		0.0229				0.0341				0.0032				0.0026				
		0.0019				0.0065												
						0.0008												
Cumulative % of Total Variation		58.6				58.4				76.5				69.1				
		90.3				85.8				91.9				93.8				
		97.5				98.5				97.1				99.5				
		99.6				99.4				99.9				99.9				
		100				99.9				100				100				
		100				100												
PCs retained for 99% of variation		PC1	PC2	PC3	PC4	PC1	PC2	PC3	PC4	PC1	PC2	PC3	PC4	PC1	PC2	PC3		
Rotated Eigenvectors ( $\mathbf{v}^{p'}$ )		Obj1	-0.5629	0.1159	0.0583	-0.2692	-0.4472	-0.1393	-0.2825	-0.2806								
		Obj2					0.5453	-0.0545	-0.0180	0.0671	-0.6673	-0.0068	0.0128	-0.0603	-0.0713	0.0021	0.6535	
		Obj3	-0.0980	0.0501	-0.0890	-0.8433	-0.0603	0.6565	-0.2382	-0.1855	0.7448	-0.0069	0.0130	-0.0610	-0.0634	-0.0002	-0.7525	
		Obj4					0.0275	0.0330	0.0960	-0.8943								
		Obj5					-0.0650	-0.0317	0.8975	-0.0999	-0.0052	0.0004	-0.0008	-0.9963	-0.6550	-0.0484	0.0622	
		Obj6	0.7477	0.1450	-0.0093	-0.1095	0.7025	-0.0184	-0.1132	-0.2256	0.0006	1.0000	0.0001	-0.0004	0.7496	-0.0489	0.0528	
		Obj7	0.1761	0.7683	0.1497	-0.2026												
		Obj8																
		Obj9	0.0675	-0.0354	-0.9518	-0.0711						0.0011	-0.0001	-0.9998	-0.0008	-0.0051	-0.9976	-0.0041
		Obj10	0.2810	-0.6095	0.2455	-0.3979	0.0230	0.7378	0.1886	0.1471								
PCs retained for 95% of variation		PC1	PC2	PC3		PC1	PC2	PC3		PC1	PC2	PC3		PC1	PC2	PC3		
Rotated Eigenvectors ( $\mathbf{v}^{p'}$ )		Obj1	-0.6156	0.1149	0.0643		-0.5671	-0.0897	-0.1956									
		Obj2					0.5033	-0.0501	-0.0475		-0.6570	-0.0468	0.0279		-0.0713	0.0021	0.6535	
		Obj3	-0.5664	-0.0099	-0.0155		-0.1337	0.6880	-0.1684		0.7430	-0.0483	0.0154		-0.0634	-0.0002	-0.7525	
		Obj4					-0.3694	0.1590	0.3188									
		Obj5					0.0193	-0.0713	0.8940		0.0852	-0.7006	0.1755		-0.6550	-0.0484	0.0622	
		Obj6	0.5474	0.1022	0.0433		0.4964	0.0378	-0.0682		0.0912	0.7104	0.1767		0.7496	-0.0489	0.0528	
		Obj7	0.0225	0.7305	0.2256													
		Obj8																
		Obj9	0.0007	0.0110	-0.9382							0.0250	0.0005	-0.9680		-0.0051	-0.9976	-0.0041
		Obj10	0.0109	-0.6652	0.2502		0.1548	0.6959	0.1603									
PCs retained for 85% of variation		PC1	PC2			PC1	PC2			PC1	PC2			PC1	PC2			
Rotated Eigenvectors ( $\mathbf{v}^{p'}$ )		Obj1	-0.5169	0.0461		-0.4615	-0.0897											
		Obj2					0.4869	-0.0530			-0.5337	-0.1036			-0.2560	0.4832		
		Obj3	-0.5104	-0.0572			-0.0591	0.6863			0.5737	0.0234			0.1818	-0.5288		
		Obj4					-0.4571	0.1649										
		Obj5					-0.3001	-0.0596			-0.0472	-0.6608			-0.5593	0.1297		
		Obj6	0.5067	0.1408			0.4886	0.0347			-0.0322	0.7327			0.5459	-0.1296		
		Obj7	0.1271	0.6761														
		Obj8																
		Obj9	-0.4297	0.1822								0.6187	-0.1233			-0.5392	-0.6733	
		Obj10	0.1212	-0.6961			0.0940	0.6973										

## G.2 PCA and Potential Objective Reduction - 2nd Stage

For the final populations from the optimisations of the re-formulated objectives in Clusters 2 and 3, FAST-MCD was used to determine robust correlation matrices, on which to base PCA and as before, the proportion of the population retained was 95%. Subsequently, PCA was conducted and the results are shown in Table G.3. The grey-shaded eigenvector coefficients indicate the retained objectives resulting from application of the objective reduction rules.

A similar process to that in Section 6.6.2 was used to explore what effect varying the cumulative percentage of total variation had on the number of objectives retained. In order to achieve this, the Objective reduction Rules OR3a)-3c) were applied, the results of which are shown in Table G.4. A reasonable objective reduction was achieved in Cluster 3 with a 99% threshold for the cumulative percentage of variation, but no objective reduction was achieved for Clusters: 2, 4\_1 and 4\_2. With a 95% threshold, there was a small objective reduction in Cluster 2, but none in Clusters: 4\_1 and 4\_2. With an 85% threshold, a reasonable objective reduction was achieved for Clusters 2, 4\_1 and 4\_2. However, for Cluster 3, an 85% threshold was considered to give too drastic an objective reduction in one go and so, it was decided to adhere to using a 99% threshold for this cluster. For this case study a policy of progressively, gradually reducing the number of objectives was adopted, albeit that this approach may involve more objective reduction stages and therefore take longer.

The application of the Objective reduction Rules using the notation is described

Table G.4: 2nd Stage retained objectives from objective reduction process for varying thresholds of cumulative percentage of total variation. The objective reduction selected per cluster is shown highlighted in grey. The number of retained objectives from the first stage is shown in brackets for each cluster.

Threshold for cumulative % of total variation	No. of objectives retained			
	Cluster 2 (6)	Cluster 3 (7)	Cluster 4_1 (5)	Cluster 4_2 (5)
99	6	5	5	5
95	5	4	5	5
85	4	3	4	4



below.

### G.2.1 Retaining PCs to account for 99% of the Total Variation

#### Objective Reduction Rules applied within Cluster 2.

1. Applying OR1,  $\lambda_1/\lambda_{10} = 1849.2$ , so there was not sufficient evidence of a near-linear dependency.
2. Applying OR2,  $t_2 = 99\%$ , *i.e.* the first four PCs were retained, which accounted for approximately 99% of the cumulative percentage total variation.
3. Applying OR3a),  $\theta_6 = 0.4083$ .
4. Applying OR3c) to PC1,  ${}_1A_-(1) \wedge {}_1A_+(6) \Rightarrow \{1, 6\} \subseteq I$ .
5. Applying OR3c) to PC2,  ${}_2A_-(10) \wedge {}_2A_+(7) \Rightarrow \{7, 10\} \subseteq I$ .
6. Applying OR3b) to PC3,  ${}_3A_-(9) \Rightarrow \{9\} \subseteq I$ .
7. Applying OR3b) to PC4,  ${}_4A_-(3) \Rightarrow \{3\} \subseteq I$ .

In summary, all six objectives were retained in Cluster 2, *i.e.* no further objective reduction was achieved.

#### Objective Reduction Rules applied within Cluster 3.

1. Applying OR1,  $\lambda_1/\lambda_{10} = 5107.1$ , so there was not sufficient evidence of a near-linear dependency.
2. Applying OR2,  $t_2 = 99\%$ , *i.e.* the first four PCs were retained, which accounted for approximately 99% of the cumulative percentage total variation.
3. Applying OR3a),  $\theta_7 = 0.3780$ .
4. Applying OR3c) to PC1,  ${}_1A_-(1) \wedge {}_1A_+(2, 6) \wedge (r_6 > r_2) \Rightarrow \{1, 6\} \subseteq I$ .
5. Applying OR3b) to PC2,  ${}_2A_+(3, 10) \wedge (r_{10} > r_3) \Rightarrow \{10\} \subseteq I$ .
6. Applying OR3b) to PC3,  ${}_3A_+(5) \Rightarrow \{5\} \subseteq I$ .
7. Applying OR3b) to PC4,  ${}_4A_-(4) \Rightarrow \{4\} \subseteq I$ .

In summary, five objectives were retained in Cluster 3:

- neg\_25\_INT
- cyc\_612\_RMS
- F\_model
- Peak\_Flare\_Speed
- sf\_cyc\_612\_INT\_abs

#### Objective Reduction Rules applied within Cluster 4\_1.

1. Applying OR1,  $\lambda_1/\lambda_{10} = 1195.2$ , so there was not sufficient evidence of a near-linear dependency.
2. Applying OR2,  $t_2 = 99\%$ , *i.e.* the first four PCs were retained, which accounted for approximately 99% of the cumulative percentage total variation.
3. Applying OR3a),  $\theta_5 = 0.4472$ .
4. Applying OR3c) to PC1,  ${}_1A_-(2) \wedge {}_1A_+(3) \Rightarrow \{2, 3\} \subseteq I$ .
5. Applying OR3b) to PC2,  ${}_2A_+(6) \Rightarrow \{6\} \subseteq I$ .
6. Applying OR3b) to PC3,  ${}_3A_-(8) \Rightarrow \{8\} \subseteq I$ .
7. Applying OR3b) to PC4,  ${}_4A_-(5) \Rightarrow \{5\} \subseteq I$ .

In summary, all five objectives were retained in Cluster 4\_1, *i.e.* no further objective reduction was achieved.

#### Objective Reduction Rules applied within Cluster 4\_2.

1. Applying OR1,  $\lambda_1/\lambda_{10} = 1328$ , so there was not sufficient evidence of a near-linear dependency.
2. Applying OR2,  $t_2 = 99\%$ , *i.e.* the first three PCs were retained, which accounted for approximately 99% of the cumulative percentage total variation.
3. Applying OR3a),  $\theta_5 = 0.4472$ .

4. Applying OR3c) to PC1,  ${}_1A_-(5) \wedge {}_1A_+(6) \Rightarrow \{5, 6\} \subseteq I$ .
5. Applying OR3b) to PC2,  ${}_2A_-(8) \Rightarrow \{8\} \subseteq I$ .
6. Applying OR3c) to PC3,  ${}_3A_-(3) \wedge {}_3A_+(2) \Rightarrow \{2, 3\} \subseteq I$ .

In summary, all five objectives were retained in Cluster 4\_2, *i.e.* no further objective reduction was achieved.

## G.2.2 Retaining PCs to account for 95% of the Total Variation

### Objective Reduction Rules applied within Cluster 2.

1. Applying OR1,  $\lambda_1/\lambda_{10} = 1849.2$ , so there was not sufficient evidence of a near-linear dependency.
2. Applying OR2,  $t_2 = 95\%$ , *i.e.* the first three PCs were retained, which accounted for approximately 95% of the cumulative percentage total variation.
3. Applying OR3a),  $\theta_6 = 0.4083$ .
4. Applying OR3c) to PC1,  ${}_1A_-(1, 3) \wedge {}_1A_+(6) \wedge (r_1 > r_3) \Rightarrow \{1, 6\} \subseteq I$ .
5. Applying OR3c) to PC2,  ${}_2A_-(10) \wedge {}_2A_+(7) \Rightarrow \{7, 10\} \subseteq I$ .
6. Applying OR3b) to PC2,  ${}_3A_-(9) \Rightarrow \{9\} \subseteq I$ .

In summary, five objectives were retained in Cluster 2:

- neg\_25\_INT
- Peak\_Flare\_Speed
- sf\_cyc\_25\_RMS\_abs
- sf\_cyc\_25\_INT\_abs
- sf\_cyc\_612\_INT\_abs

**Objective Reduction Rules applied within Cluster 3.**

1. Applying OR1,  $\lambda_1/\lambda_{10} = 5107.1$ , so there was not sufficient evidence of a near-linear dependency.
2. Applying OR2,  $t_2 = 95\%$ , *i.e.* the first three PCs were retained, which accounted for approximately 95% of the cumulative percentage total variation.
3. Applying OR3a),  $\theta_7 = 0.3780$ .
4. Applying OR3c) to PC1,  ${}_1A_-(1) \wedge {}_1A_+(2, 6) \wedge (r_6 > r_2) \Rightarrow \{1, 6\} \subseteq I$ .
5. Applying OR3b) to PC2,  ${}_2A_+(3, 10) \wedge (r_{10} > r_3) \Rightarrow \{10\} \subseteq I$ .
6. Applying OR3b) to PC3,  ${}_3A_+(5) \Rightarrow \{5\} \subseteq I$ .

In summary, four objectives were retained in Cluster 3:

- neg\_25\_INT
- F\_model
- Peak\_Flare\_Speed
- sf\_cyc\_612\_INT\_abs

**Objective Reduction Rules applied within Cluster 4\_1.**

1. Applying OR1,  $\lambda_1/\lambda_{10} = 1195.2$ , so there was not sufficient evidence of a near-linear dependency.
2. Applying OR2,  $t_2 = 95\%$ , *i.e.* the first three PCs were retained, which accounted for approximately 95% of the cumulative percentage total variation.
3. Applying OR3a),  $\theta_5 = 0.4472$ .
4. Applying OR3c) to PC1,  ${}_1A_-(2) \wedge {}_1A_+(3) \Rightarrow \{2, 3\} \subseteq I$ .
5. Applying OR3c) to PC2,  ${}_2A_-(5) \wedge {}_2A_+(6) \Rightarrow \{5, 6\} \subseteq I$ .
6. Applying OR3b) to PC3,  ${}_3A_-(9) \Rightarrow \{9\} \subseteq I$ .

In summary, all five objectives were retained in Cluster 4\_1, *i.e.* no further objective reduction was achieved.

**Objective Reduction Rules applied within Cluster 4\_2.**

1. Applying OR1,  $\lambda_1/\lambda_{10} = 1328$ , so there was not sufficient evidence of a near-linear dependency.
2. Applying OR2,  $t_2 = 95\%$ , *i.e.* the first three PCs were retained, which accounted for approximately 95% of the cumulative percentage total variation.
3. Applying OR3a),  $\theta_5 = 0.4472$ .
4. Applying OR3c) to PC1,  ${}_1A_-(5) \wedge {}_1A_+(6) \Rightarrow \{5, 6\} \subseteq I$ .
5. Applying OR3b) to PC2,  ${}_2A_-(9) \Rightarrow \{9\} \subseteq I$ .
6. Applying OR3c) to PC3,  ${}_3A_-(3) \wedge {}_3A_+(2) \Rightarrow \{2, 3\} \subseteq I$ .

In summary, all five objectives were retained in Cluster 4\_2, *i.e.* no further objective reduction was achieved.

**G.2.3 Retaining PCs to account for 85% of the Total Variation****Objective Reduction Rules applied within Cluster 2.**

1. Applying OR1,  $\lambda_1/\lambda_{10} = 1849.2$ , so there was not sufficient evidence of a near-linear dependency.
2. Applying OR2,  $t_2 = 85\%$ , *i.e.* the first two PCs were retained, which accounted for approximately 85% of the cumulative percentage total variation.
3. Applying OR3a),  $\theta_6 = 0.4083$ .
4. Applying OR3c) to PC1,  ${}_1A_-(1, 3, 9) \wedge {}_1A_+(6) \wedge (r_1 > r_9 > r_3) \Rightarrow \{1, 6\} \subseteq I$ .
5. Applying OR3c) to PC2,  ${}_2A_-(7) \wedge {}_2A_+(10) \Rightarrow \{7, 10\} \subseteq I$ .

In summary, four objectives were retained in Cluster 2:

- neg\_25\_INT
- Peak\_Flare\_Speed
- sf\_cyc\_25\_INT\_abs
- sf\_cyc\_612\_INT\_abs

**Objective Reduction Rules applied within Cluster 3.**

1. Applying OR1,  $\lambda_1/\lambda_{10} = 5107.1$ , so there was not sufficient evidence of a near-linear dependency.
2. Applying OR2,  $t_2 = 85\%$ , *i.e.* the first two PCs were retained, which accounted for approximately 85% of the cumulative percentage total variation.
3. Applying OR3a),  $\theta_7 = 0.3780$ .
4. Applying OR3c) to PC1,  
 ${}_1A_-(1, 4) \wedge {}_1A_+(2, 6) \wedge (r_1 > r_4) \wedge (r_6 > r_2) \Rightarrow \{1, 6\} \subseteq I$ .
5. Applying OR3b) to PC2,  ${}_2A_+(3, 10) \wedge (r_{10} > r_3) \Rightarrow \{10\} \subseteq I$ .

In summary, three objectives were retained in Cluster 3:

- neg\_25\_INT
- Peak\_Flare\_Speed
- sf\_cyc\_612\_INT\_abs

**Objective Reduction Rules applied within Cluster 4.1.**

1. Applying OR1,  $\lambda_1/\lambda_{10} = 1195.2$ , so there was not sufficient evidence of a near-linear dependency.
2. Applying OR2,  $t_2 = 85\%$ , *i.e.* the first two PCs were retained, which accounted for approximately 85% of the cumulative percentage total variation.
3. Applying OR3a),  $\theta_5 = 0.4472$ .
4. Applying OR3c) to PC1,  ${}_1A_-(2) \wedge {}_1A_+(3, 8) \wedge (r_3 > r_8) \Rightarrow \{2, 3\} \subseteq I$ .
5. Applying OR3c) to PC2,  ${}_2A_-(5) \wedge {}_2A_+(6) \Rightarrow \{5, 6\} \subseteq I$ .

In summary, four objectives were retained in Cluster 4.1:

- neg\_612\_INT
- cyc\_25\_RMS
- F\_model
- Peak\_Flare\_Speed

**Objective Reduction Rules applied within Cluster 4\_2.**

1. Applying OR1,  $\lambda_1/\lambda_{10} = 1328$ , so there was not sufficient evidence of a near-linear dependency.
2. Applying OR2,  $t_2 = 85\%$ , *i.e.* the first two PCs were retained, which accounted for approximately 85% of the cumulative percentage total variation.
3. Applying OR3a),  $\theta_5 = 0.4472$ .
4. Applying OR3c) to PC1,  ${}_1A_-(5, 8) \wedge {}_1A_+(6) \wedge (r_5 > r_8) \Rightarrow \{5, 6\} \subseteq I$ .
5. Applying OR3c) to PC2,  ${}_2A_-(3, 8) \wedge {}_2A_+(2) \wedge (r_3 > r_8) \Rightarrow \{3, 8\} \subseteq I$ .

In summary, four objectives were retained in Cluster 4\_2:

- cyc\_25\_RMS
- F\_model
- Peak\_Flare\_Speed
- sf\_cyc\_612\_RMS\_abs





## Appendix H

# Detailed Results from Objective Reduction Process - 3rd Stage

This Appendix provides details of the Stage 3 objective reduction processes summarised in Section 6.8.3 and depicted in the flowchart in Figure 6.8.

### H.1 PCA and Potential Objective Reduction - 3rd Stage

For the final populations from the further optimisations with reduced objectives Stage 3, FAST-MCD was used to determine robust correlation matrices, on which to base PCA and as before, the proportion of the population retained was 95%. Subsequently, PCA was conducted and the results are shown in Table H.1. The grey-shaded eigenvector coefficients indicate the retained objectives resulting from application of the objective reduction rules.

The application of the Objective reduction Rules using the notation is described below.

#### H.1.1 Retaining PCs to account for 99% of the Total Variation

##### Objective Reduction Rules applied within Cluster 2.

1. Applying OR1,  $\lambda_1/\lambda_{10} = 93.1$ , so there was not sufficient evidence of a near-linear dependency.
2. Applying OR2,  $t_2 = 99\%$ , *i.e.* the first three PCs were retained, which accounted for approximately 99% of the cumulative percentage total variation.

Table H.1: Stage 3 results from PCA applied to optimisations of reduced objectives within Clusters 2 to 4. Note missing rows in the tables of eigenvectors indicate discarded objectives.

		Cluster 2			Cluster 3			Cluster 4.1			Cluster 4.2		
PC Eigenvalues ( $\lambda_p$ )		2.3825			3.4125			3.1247			3.2291		
		1.5438			0.9990			0.7381			0.6898		
		0.0481			0.5624			0.1342			0.0801		
		0.0256			0.0225			0.0030			0.0010		
Cumulative % of Total Variation		59.6			68.3			78.1			80.7		
		98.2			88.2			96.6			98.0		
		99.4			99.5			99.9			100		
		100			99.9			100			100		
PCs retained for 99% of variation		PC1	PC2	PC3	PC1	PC2	PC3	PC1	PC2	PC3	PC1	PC2	PC3
Rotated Eigenvectors ( $\mathbf{v}^r$ )	Obj1	0.0590	0.6951	-0.1856	-0.6326	0.1605	-0.1211	-0.6847 0.7288	-0.0068 -0.0068	0.0448 0.0450	-0.0117	0.0135	-0.9816
	Obj2	0.0587 -0.0118	-0.7189 0.0030	-0.1835 -0.9653	-0.5251 -0.0238	-0.0959 -0.9810	0.1087 -0.0156						
	Obj3												
	Obj4												
	Obj5												
	Obj6												
	Obj7												
	Obj8												
	Obj9												
	Obj10												
PCs retained for 95% of variation								PC1	PC2		PC1	PC2	PC3
Rotated Eigenvectors ( $\mathbf{v}^r$ )	Obj1	-0.0938	0.6855		-0.6326	0.1605	-0.1211	-0.6850 0.7284	-0.0368 -0.0357		-0.4063	0.3683	
	Obj2	-0.0929 -0.7084	-0.7266 -0.0303		-0.5251 -0.0238	-0.0959 -0.9810	0.1087 -0.0156						
	Obj3												
	Obj4												
	Obj5												
	Obj6												
	Obj7												
	Obj8												
	Obj9												
	Obj10												0.6934
PCs retained for 85% of variation								PC1	PC2		PC1	PC2	
Rotated Eigenvectors ( $\mathbf{v}^r$ )	Obj1	-0.0938	0.6855		-0.6313	0.1841	-0.6850 0.7284	-0.0368 -0.0357		-0.4063	0.3683		
	Obj2	-0.0929 -0.7084	-0.7266 -0.0303		-0.5204 0.0784	-0.1555 -0.7818							
	Obj3												
	Obj4												
	Obj5												
	Obj6												
	Obj7												
	Obj8												
	Obj9												
	Obj10											0.6934	-0.0356

3. Applying OR3a),  $\theta_4 = 0.5$ .
4. Applying OR3b) to PC1,  ${}_1A_+(10) \Rightarrow \{10\} \subseteq I$ .
5. Applying OR3c) to PC2,  ${}_2A_-(6) \wedge {}_2A_+(1) \Rightarrow \{1, 6\} \subseteq I$ .
6. Applying OR3b) to PC3,  ${}_3A_+(7) \Rightarrow \{7\} \subseteq I$ .

In summary, all four objectives were retained in Cluster 2, *i.e.* no further objective reduction was achieved.

### Objective Reduction Rules applied within Cluster 3.

1. Applying OR1,  $\lambda_1/\lambda_{10} = 947.9$ , so there was not sufficient evidence of a near-linear dependency.
2. Applying OR2,  $t_2 = 99\%$ , *i.e.* the first three PCs were retained, which accounted for approximately 99% of the cumulative percentage total variation.
3. Applying OR3a),  $\theta_5 = 0.4472$ .
4. Applying OR3c) to PC1,  ${}_1A_-(1) \wedge {}_1A_+(6) \Rightarrow \{1, 6\} \subseteq I$ .
5. Applying OR3b) to PC2,  ${}_2A_-(5) \Rightarrow \{5\} \subseteq I$ .
6. Applying OR3b) to PC3,  ${}_3A_-(10) \Rightarrow \{10\} \subseteq I$ .

In summary, four objectives were retained in Cluster 3:

- neg\_25\_INT
- F\_model
- Peak\_Flare\_Speed
- sf\_cyc\_612\_INT\_abs

### Objective Reduction Rules applied within Cluster 4\_1.

1. Applying OR1,  $\lambda_1/\lambda_{10} = 1041.6$ , so there was not sufficient evidence of a near-linear dependency.
2. Applying OR2,  $t_2 = 99\%$ , *i.e.* the first three PCs were retained, which accounted for approximately 99% of the cumulative percentage total variation.

3. Applying OR3a),  $\theta_4 = 0.5$ .
4. Applying OR3c) to PC1,  ${}_1A_-(2) \wedge {}_1A_+(3) \Rightarrow \{2, 3\} \subseteq I$ .
5. Applying OR3b) to PC2,  ${}_2A_-(5) \Rightarrow \{5\} \subseteq I$ .
6. Applying OR3b) to PC3,  ${}_3A_-(6) \Rightarrow \{6\} \subseteq I$ .

In summary, four objectives were retained in Cluster 4.1:

- neg\_612\_INT
- cyc\_25\_RMS
- F\_model
- Peak\_Flare\_Speed

#### Objective Reduction Rules applied within Cluster 4.2.

1. Applying OR1,  $\lambda_1/\lambda_{10} = 3229.1$ , so there was not sufficient evidence of a near-linear dependency.
2. Applying OR2,  $t_2 = 99\%$ , *i.e.* the first three PCs were retained, which accounted for approximately 99% of the cumulative percentage total variation.
3. Applying OR3a),  $\theta_4 = 0.5$ .
4. Applying OR3c) to PC1,  ${}_1A_-(5) \wedge {}_1A_+(6) \Rightarrow \{5, 6\} \subseteq I$ .
5. Applying OR3b) to PC2,  ${}_2A_-(8) \Rightarrow \{8\} \subseteq I$ .
6. Applying OR3b) to PC3,  ${}_3A_-(2) \Rightarrow \{2\} \subseteq I$ .

In summary, four objectives were retained in Cluster 4.2:

- neg\_612\_INT
- F\_model
- Peak\_Flare\_Speed
- sf\_cyc\_612\_RMS\_abs

### H.1.2 Retaining PCs to account for 95% of the Total Variation

#### Objective Reduction Rules applied within Cluster 2.

1. Applying OR1,  $\lambda_1/\lambda_{10} = 93.1$ , so there was not sufficient evidence of a near-linear dependency.
2. Applying OR2,  $t_2 = 95\%$ , *i.e.* the first two PCs were retained, which accounted for approximately 95% of the cumulative percentage total variation.
3. Applying OR3a),  $\theta_4 = 0.5$ .
4. Applying OR3c) to PC1,  ${}_1A_+(7) \wedge {}_1A_+(10) \Rightarrow \{7, 10\} \subseteq I$ .
5. Applying OR3c) to PC2,  ${}_2A_-(6) \wedge {}_2A_+(1) \Rightarrow \{1, 6\} \subseteq I$ .

In summary, all four objectives were retained in Cluster 2, *i.e.* no further objective reduction was achieved.

#### Objective Reduction Rules applied within Cluster 3.

1. Applying OR1,  $\lambda_1/\lambda_{10} = 947.9$ , so there was not sufficient evidence of a near-linear dependency.
2. Applying OR2,  $t_2 = 95\%$ , *i.e.* the first three PCs were retained, which accounted for approximately 95% of the cumulative percentage total variation.
3. Applying OR3a),  $\theta_5 = 0.4472$ .
4. Applying OR3c) to PC1,  ${}_1A_-(1) \wedge {}_1A_+(6) \Rightarrow \{1, 6\} \subseteq I$ .
5. Applying OR3b) to PC2,  ${}_2A_-(5) \Rightarrow \{5\} \subseteq I$ .
6. Applying OR3b) to PC3,  ${}_3A_-(10) \Rightarrow \{10\} \subseteq I$ .

In summary, four objectives were retained in Cluster 3:

- neg\_25\_INT
- F\_model
- Peak\_Flare\_Speed
- sf\_cyc\_612\_INT\_abs

**Objective Reduction Rules applied within Cluster 4\_1.**

1. Applying OR1,  $\lambda_1/\lambda_{10} = 1041.6$ , so there was not sufficient evidence of a near-linear dependency.
2. Applying OR2,  $t_2 = 95\%$ , *i.e.* the first two PCs were retained, which accounted for approximately 95% of the cumulative percentage total variation.
3. Applying OR3a),  $\theta_4 = 0.5$ .
4. Applying OR3c) to PC1,  ${}_1A_-(2) \wedge {}_1A_+(3) \Rightarrow \{2, 3\} \subseteq I$ .
5. Applying OR3c) to PC2,  ${}_2A_-(5) \wedge {}_1A_+(6) \Rightarrow \{5, 6\} \subseteq I$ .

In summary, all four objectives were retained in Cluster 4\_1, *i.e.* no further objective reduction was achieved.

**Objective Reduction Rules applied within Cluster 4\_2.**

1. Applying OR1,  $\lambda_1/\lambda_{10} = 3229.1$ , so there was not sufficient evidence of a near-linear dependency.
2. Applying OR2,  $t_2 = 95\%$ , *i.e.* the first two PCs were retained, which accounted for approximately 95% of the cumulative percentage total variation.
3. Applying OR3a),  $\theta_4 = 0.5$ .
4. Applying OR3c) to PC1,  ${}_1A_-(5) \wedge {}_1A_+(6) \Rightarrow \{5, 6\} \subseteq I$ .
5. Applying OR3b) to PC2,  ${}_2A_-(8) \Rightarrow \{8\} \subseteq I$ .

In summary, three objectives were retained in Cluster 4\_2:

- F\_model
- Peak\_Flare\_Speed
- sf\_cyc\_612\_RMS\_abs

### H.1.3 Retaining PCs to account for 85% of the Total Variation

#### Objective Reduction Rules applied within Cluster 2.

1. Applying OR1,  $\lambda_1/\lambda_{10} = 93.1$ , so there was not sufficient evidence of a near-linear dependency.
2. Applying OR2,  $t_2 = 85\%$ , *i.e.* the first two PCs were retained, which accounted for approximately 85% of the cumulative percentage total variation.
3. Applying OR3a),  $\theta_4 = 0.5$ .
4. Applying OR3c) to PC1,  ${}_1A_+(7) \wedge {}_1A_+(10) \Rightarrow \{7, 10\} \subseteq I$ .
5. Applying OR3c) to PC2,  ${}_2A_-(6) \wedge {}_2A_+(1) \Rightarrow \{1, 6\} \subseteq I$ .

In summary, all four objectives were retained in Cluster 2, *i.e.* no further objective reduction was achieved.

#### Objective Reduction Rules applied within Cluster 3.

1. Applying OR1,  $\lambda_1/\lambda_{10} = 947.9$ , so there was not sufficient evidence of a near-linear dependency.
2. Applying OR2,  $t_2 = 85\%$ , *i.e.* the first two PCs were retained, which accounted for approximately 85% of the cumulative percentage total variation.
3. Applying OR3a),  $\theta_5 = 0.4472$ .
4. Applying OR3c) to PC1,  ${}_1A_-(1) \wedge {}_1A_+(6) \Rightarrow \{1, 6\} \subseteq I$ .
5. Applying OR3b) to PC2,  ${}_2A_-(5) \Rightarrow \{5\} \subseteq I$ .
6. For PC2, the sign of the eigenvector coefficient for Objective 10 was mistaken as being positive (it is in fact, negative) and so OR3c) was erroneously applied to PC2,  ${}_2A_-(5) \wedge {}_2A_+(10) \Rightarrow \{5, 10\} \subseteq I$ . (The correct rule to apply to PC2 was OR3b),  ${}_2A_-(5) \Rightarrow \{5\} \subseteq I$  and so, only three objectives should have been retained).

In summary (albeit in error), four objectives were retained in Cluster 3:

- neg\_25\_INT

- F\_model
- Peak\_Flare\_Speed
- sf\_cyc\_612\_INT\_abs

#### Objective Reduction Rules applied within Cluster 4\_1.

1. Applying OR1,  $\lambda_1/\lambda_{10} = 1041.6$ , so there was not sufficient evidence of a near-linear dependency.
2. Applying OR2,  $t_2 = 85\%$ , *i.e.* the first two PCs were retained, which accounted for approximately 85% of the cumulative percentage total variation.
3. Applying OR3a),  $\theta_4 = 0.5$ .
4. Applying OR3c) to PC1,  ${}_1A_-(2) \wedge {}_1A_+(3) \Rightarrow \{2, 3\} \subseteq I$ .
5. Applying OR3c) to PC2,  ${}_2A_-(5) \wedge {}_1A_+(6) \Rightarrow \{5, 6\} \subseteq I$ .

In summary, all four objectives were retained in Cluster 4\_1, *i.e.* no further objective reduction was achieved.

#### Objective Reduction Rules applied within Cluster 4\_2.

1. Applying OR1,  $\lambda_1/\lambda_{10} = 3229.1$ , so there was not sufficient evidence of a near-linear dependency.
2. Applying OR2,  $t_2 = 85\%$ , *i.e.* the first two PCs were retained, which accounted for approximately 85% of the cumulative percentage total variation.
3. Applying OR3a),  $\theta_4 = 0.5$ .
4. Applying OR3c) to PC1,  ${}_1A_-(5) \wedge {}_1A_+(6) \Rightarrow \{5, 6\} \subseteq I$ .
5. Applying OR3b) to PC2,  ${}_2A_-(8) \Rightarrow \{8\} \subseteq I$ .

In summary, three objectives were retained in Cluster 4\_2:

- F\_model
- Peak\_Flare\_Speed
- sf\_cyc\_612\_RMS\_abs



# Appendix I

## Detailed Results from Objective Reduction Process - 4th Stage

This Appendix provides details of the Stage 4 objective reduction processes summarised in Section 6.9.3 and depicted in the flowchart in Figure 6.9.

### I.1 PCA and Potential Objective Reduction - 4th Stage

#### I.1.1 Retaining PCs to account for 95% of the Total Variation

##### Objective Reduction Rules applied within Cluster 3\_1.

1. Applying OR1,  $\lambda_1/\lambda_{10} = 286.2$ , so there was not sufficient evidence of a near-linear dependency.
2. Applying OR2,  $t_2 = 95\%$ , *i.e.* the first three PCs were retained, which accounted for approximately 95% of the cumulative percentage total variation.
3. Applying OR3a),  $\theta_4 = 0.5$ .
4. Applying OR3c) to PC1,  ${}_1A_+(1) \wedge {}_1A_+(6) \Rightarrow \{1, 6\} \subseteq I$ .
5. Applying OR3b) to PC2,  ${}_2A_-(5) \Rightarrow \{5\} \subseteq I$ .
6. Applying OR3b) to PC3,  ${}_3A_+(10) \Rightarrow \{10\} \subseteq I$ .

In summary, all four objectives were retained in Cluster 3\_1, *i.e.* no further objective reduction was achieved.

Table I.1: Stage 4 results from PCA applied to optimisations of reduced objectives within Cluster 3. Note missing rows in the tables of eigenvectors indicate discarded objectives.

		Cluster 3_1		
PC Eigenvalues ( $\lambda_p$ )		2.5181	0.8974	0.5757
		0.0088		
Cumulative % of Total Variation		63.0	85.4	99.8
		100		
PCs retained for 95% of variation		PC1	PC2	PC3
Rotated Eigenvectors ( $\mathbf{v}^{p'}$ )	Obj1	-0.7548	0.1194	-0.0633
	Obj2			
	Obj3			
	Obj4			
	Obj5	-0.0111	-0.9854	-0.0079
	Obj6	0.6558	0.1208	-0.0638
	Obj7			
	Obj8			
	Obj9			
	Obj10	-0.0061	0.0075	0.9959
PCs retained for 85% of variation		PC1	PC2	
Rotated Eigenvectors ( $\mathbf{v}^{p'}$ )	Obj1	-0.7516	0.1155	
	Obj2			
	Obj3			
	Obj4			
	Obj5	0.1003	-0.7881	
	Obj6	0.6411	0.1508	
	Obj7			
	Obj8			
	Obj9			
	Obj10	-0.1182	-0.5855	

### I.1.2 Retaining PCs to account for 85% of the Total Variation

#### Objective Reduction Rules applied within Cluster 3\_1.

1. Applying OR1,  $\lambda_1/\lambda_{10} = 286.2$ , so there was not sufficient evidence of a near-linear dependency.
2. Applying OR2,  $t_2 = 85\%$ , *i.e.* the first two PCs were retained, which accounted for approximately 85% for the cumulative percentage of total variation.
3. Applying OR3a),  $\theta_4 = 0.5$ .
4. Applying OR3c) to PC1,  ${}_1A_+(1) \wedge {}_1A_+(6) \Rightarrow \{1, 6\} \subseteq I$ .
5. For PC2, the sign of the eigenvector coefficient for Objective 10 was mistaken again as being positive (it is in fact, negative) and so OR3c) was erroneously applied to PC2,  ${}_2A_-(5) \wedge {}_2A_+(10) \Rightarrow \{5, 10\} \subseteq I$ . (The correct rule to apply to PC2 was OR3b),  ${}_2A_-(5) \Rightarrow \{5\} \subseteq I$  and so, only three objectives should have been retained).

In summary (albeit in error), four objectives were retained in Cluster 3\_1, *i.e.* no further objective reduction was achieved.



# References

- Adra, S., Dodd, T. J., Griffin, I. A. and Fleming, P. J.: 2009, A Convergence Acceleration Operator for Multiobjective Optimisation, *IEEE Transactions on Evolutionary Computation* **13**(4), 825–847.
- Adra, S., Griffin, I. and Fleming, P. J.: 2007, A Comparative Study of Progressive Preference Articulation Techniques for Multiobjective Optimisation, *Evolutionary Multiobjective Optimisation, EMO 2007*.
- Agrawal, G., Lewis, K. E., Chugh, K., Huang, C., Parashar, S. and Bloebaum, C. L.: 2004, Intuitive visualization of pareto frontier for multi-objective optimization in n-dimensional performance space, *Proceedings of 10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, Albany, NY.
- Al-Kandari, N. M.: 1998, Variable Selection and Interpretation in Principal Component Analysis. Unpublished Ph.D. thesis.
- Alger, T., Hall, M. and Matthews, R. D.: 2000, Effects of swirl and tumble on in-cylinder fuel distribution in a central injected DISI engine, *Proceedings of the SAE 2000 World Congress*, number 2000-01-0533 in *Direct Injection SI Engine Technology 2000 (SP1499)*, SAE, Warrendale, PA, USA.
- Anderson, R. W., Brehob, D. D., Yang, J., Vallance, J. K. and Whiteaker, R. M.: 1996, A new direct injection spark ignition (DISI) combustion system for low emissions, *FISITA-96*. No. P0201.
- Andersson, J.: 2000, A survey of multiobjective optimization in engineering design, *Technical Report LiTH-IKP-R-1097*, Department of Mechanical Engineering, Linköping University, Linköping, Sweden.

- Asimov, D.: 1985, Grand Tour: A tool for viewing multidimensional data, *SIAM Journal of Science and Statistical Computing* **6**, 128–143.
- Athans, M. and Falb, P.: 1966, *Optimal Control*, McGraw-Hill, New York.
- Bader, J. and Zitzler, E.: 2008, Hype: An algorithm for fast hypervolume-based many-objective optimization, *Technical Report TIK Report 286*, Computer Engineering and Networks Laboratory (TIK), ETH Zurich.
- Bellman, R.: 1961, *Adaptive Control Processes*, Princeton University Press, Princeton, NJ.
- Ben-Tal, A.: 1980, Characterization of Pareto and Lexicographic Optimal Solutions, in G. Fandel and T. Gal (eds), *Multiple Criteria Decision Making Theory and Application*, Vol. 177 of *Lecture Notes in Economics and Mathematical Systems*, Springer-Verlag, Berlin, pp. 1–11.
- Bentley, P. J. and Wakefield, J. P.: 1997, Finding Acceptable Pareto-Optimal Solutions using Multiobjective Genetic Algorithms, in R. R. P. K. Chawdhry and R. K. Pant (eds), *2nd On-line World Conference on Soft Computing in Design and Manufacturing (WSC2)*, Vol. Part 5, Springer Verlag London Limited, London, pp. 231–240.
- Berkhin, P.: 2002, Survey of Clustering Data Mining Techniques, *Technical report*, Accrue Software.
- Bielaczyc, P. and Merkisz, J.: 1997, Exhaust emission from passenger cars during engine cold start and warm-up, *SAE International Congress & Exposition*, SAE International, Detroit, Michigan.
- Blasco, X., Herrero, J., Sanchis, J. and Martnez, M.: 2008, A new graphical visualization of n-dimensional pareto front for decision-making in multiobjective optimization, *Information Sciences* **178**(20), 3908–3924.
- Box, G. E. P. and Jenkins, G. M.: 1970, *Time series analysis: forecasting and control*, Holden-Day, London.

- Branke, J. and Deb, K.: 2004, Integrating User Preferences into Evolutionary Multi-Objective Optimization, *Technical Report 2004004*, Indian Institute of Technology, Kanpur, India.
- Branke, J., Kaußler, T. and Schmeck, H.: 2001, Guidance in Evolutionary Multi-Objective Optimization, *Advances in Engineering Software* **32**, 499–507.
- Branke, J., Schmeck, H., Deb, K. and Reddy S, M.: 2004, Parallelizing multi-objective evolutionary algorithms: cone separation, *2004 Congress on Evolutionary Computation, CEC 2004*, Vol. 2, IEEE Service Center, Portland, Oregon, USA, pp. 1952–1957.
- Brockhoff, D. and Zitzler, E.: 2006, Are All Objectives Necessary? On Dimensionality Reduction in Evolutionary Multiobjective Optimization, *Parallel Problem Solving from Nature - PPSN IX*, Vol. 4193/2006 of *Lecture Notes in Computer Science*, Springer Berlin / Heidelberg, pp. 533–542.
- Brockhoff, D. and Zitzler, E.: 2009, Objective Reduction in Evolutionary Multiobjective Optimization: Theory and Applications, *Evolutionary Computation* **17**(2), 135–166.
- Brockwell, P. J. and Davis, R. A.: 1991, *Time Series: Theory and Methods*, 2nd edn, Springer.
- Brogan, M. S., Clark, A. D. and Brisley, R. J.: 1998, Recent progress in NO<sub>x</sub> trap technology, *Proceedings of the SAE International Congress and Exposition*, number 980933 in *SAE Technical Paper*, SAE, Warrendale, PA, USA.
- Brooks, T., Lumsden, G. and Blaxhill, H.: 2005, Improving Base Engine Calibrations for Diesel Vehicles through the Use of DoE and Optimization Techniques, *Powertrain & Fluid Systems Conference and Exhibition*, SAE Technical Paper Series, SAE International, San Antonio, Texas, USA. SAE paper no.: 2005-01-3833.
- Brunt, M. F. J. and Emtage, A. L.: 1996, Evaluation of imep routines and analysis errors, SAE International. SAE paper no.: 960609.
- Carreira-Perpinan, M. A.: 1997, A Review of Dimension Reduction Techniques, *Technical Report CS-96-09*, Dept. of Computer Science, University of Sheffield.

- Cary, M.: 2003, *A Model Based Engine Calibration Methodology for a Port Fuel Injection, Spark-Ignition Engine*, PhD thesis, University of Bradford.
- Cary, M.: 2007, fastmcd.m. Matlab® R2006b m-file code of FAST-MCD (Rousseeuw, 1999).
- Cary, M.: 2008, myCluster2008a.m. Matlab® R2008a m-file code of k\*-Means (Cheung, 2003).
- Cattell, R. B.: 1966, The scree test for the number of factors, *Multivariate Behavioral Research* **1**, 245–276.
- Cattell, R. B.: 1978, *The Scientific Use of Factor Analysis in Behavioral Life Sciences*, Plenum Press, New York.
- Cheung, Y. M.: 2003, k\*-Means: A New Generalized k-Means Clustering Algorithm, *Pattern Recognition Letters* **24**(15), 2883–2893.
- Chiu, P.-W. and Bloebaum, C.: 2010, Hyper-Radial Visualization (HRV) method with range-based preferences for multi-objective decision making, *Structural and Multidisciplinary Optimization* **40**(1), 97–115.
- Chiu, P.-W. and Bloebaum, C. L.: 2008, Hyper-Radial Visualization (HRV) for decision-making in multi-objective optimization, *46th AIAA Aerospace Sciences Meeting and Exhibit*. AIAA-2008-907.
- Cho, S. and Jung, H.-S.: 2003, Design sensitivity analysis and topology optimization of displacement-loaded non-linear structures, *Computer Methods in Applied Mechanics and Engineering* **192**(22-24), 2539–2553.
- Cleveland, W. S.: 1993, *Visualizing Data*, Hobart Press, Summit, New Jersey.
- Coello, C. A. C.: 2000, Handling preferences in evolutionary multiobjective optimization: a survey, in IEEE Neural Networks Council (ed.), *Proceedings of the 2000 Congress on Evolutionary Computation (CEC 2000)*, Vol. 1, IEEE Service Center, Piscataway, New Jersey, pp. 30–37.
- Coello, C. A. C., Lamont, G. B. and Veldhuizen, D. A. V.: 2007, *Evolutionary Algorithms for Solving Multi-Objective Problems*, 2nd edn, Springer.



- Corne, D. W. and Knowles, J. D.: 2007, Techniques for highly multiobjective optimisation: some nondominated points are better than others, *Proceedings of the 2007 Genetic and Evolutionary Computation Conference - GECCO 2007*, ACM, London, England, pp. 773–780.
- Cukier, R. I., Fortuin, C. M., Shuler, K. E., Petschek, A. G. and Schailby, J. H.: 1973, Study of the sensitivity of the coupled reaction systems to uncertainties in rate coefficients: I. theory, *Journal of Chemical Physics* **59**(8), 3873–3878.
- Cvetković, D. and Coello, C. A. C.: 2004, Human Preferences and their Applications in Evolutionary MultiObjective Optimization, in Y. Jin (ed.), *Knowledge Incorporation in Evolutionary Computation*, Vol. 167 of *Studies in Fuzziness and Soft Computing*, Springer, Berlin, Heidelberg, pp. 479–502.
- Das, I. and Dennis, J.: 1998, Normal-boundary intersection: A new method for generating the Pareto surface in nonlinear multicriteria optimization problems, *SIAM Journal of Optimization* **8**, 631–657.
- Davis, T. P. and Grove, D. M.: 1992, *Engineering Quality and Experimental Design*, Longman Scientific and Technical.
- D’Azzo, J. J. and Houpis, C. H.: 1981, *Linear Control System Analysis and Design: Conventional and Modern*, 2nd edn, McGraw-Hill.
- Deb, K.: 1995, *Optimization for Engineering Design: Algorithms and Examples*, New Delhi: Prentice-Hall.
- Deb, K.: 1999, An introduction to genetic algorithms, *SADHANA - Academy Proceedings in Engineering Sciences* **24**(4), 293–315.
- Deb, K.: 2001, *Multi-Objective Optimization Using Evolutionary Algorithms*, John Wiley & Sons, Chichester, UK.
- Deb, K. and Agrawal, R. B.: 1995, Simulated Binary Crossover for Continuous Search Space, *Complex Systems* **9**(2), 115–148.
- Deb, K. and Goyal, M.: 1996, A Combined Genetic Adaptive Search (GeneAS) for Engineering Design, *Computer Science and Informatics* **26**(4), 30–45.

- Deb, K., Pratap, A., Agarwal, S. and Meyarivan, T.: 2002, A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II, *IEEE Transactions on Evolutionary Computation* **6**(2), 182–197.
- Deb, K. and Saxena, D. K.: 2005, On Finding Pareto-Optimal Solutions Through Dimensionality Reduction for Certain Large-Dimensional Multi-Objective Optimization Problems, *Technical Report 2005011*, Kanpur Genetic Algorithms Laboratory (KanGAL), Indian Institute of Technology, Kanpur, India.
- Deb, K. and Saxena, D. K.: 2006, Searching for pareto-optimal solutions through dimensionality reduction for certain large-dimensional multi-objective optimization problems, *Congress on Evolutionary Computation (CEC06)*.
- Deb, K., Thiele, L., Laumanns, M. and Zitzler, E.: 2002, Scalable Multi-Objective Optimization Test Problems, in IEEE Neural Networks Council (ed.), *Proceedings of the 2002 Congress on Evolutionary Computation (CEC 2002)*, Vol. 1, IEEE Service Center, Piscataway, New Jersey, pp. 825–830.
- Deb, K., Zope, P. and Jain, A.: 2003, Distributed computing of Pareto-optimal solutions with evolutionary algorithms, in C. M. Fonseca, P. J. Fleming, E. Zitzler, K. Deb and L. Thiele (eds), *Proceedings of the Second International Conference on Evolutionary Multi-Criterion Optimization (EMO 2003)*, Springer, Berlin, pp. 534–549.
- Delinchant, B., Wurtz, F. and Atienza, E.: 2004, Reducing sensitivity analysis time-cost of compound model, *IEEE Transactions on Magnetics* **40**(2).
- Demartines, P. and Hérault, H.: 1997, Curvilinear Component Analysis: a Self-Organising Neural Network for Non-linear Mapping of Data Sets, *IEEE Transactions on Neural Networks* **8**, 148–154.
- Drechsler, N., Drechsler, R. and Becker, B.: 2001, Multi-Objective Optimisation Based on Relation *Favour*, in E. Zitzler, K. Deb, L. Thiele, C. A. C. Coello and D. Corne (eds), *Proceedings of the First International Conference on Evolutionary Multi-Criterion Optimization (EMO 2001)*, Springer-Verlag, pp. 154–166.
- Emtage, A. L.: 2009, Run-time study on parallel moga.

- Farina, M. and Amato, P.: 2002, On the optimal solution definition for many-criteria optimization problems, in J. Keller and O. Nasraoui (eds), *Proceedings of the 2002 NAFIPS-FLINT International Conference*, IEEE Service Center, Piscataway, New Jersey, pp. 233–238.
- Ferguson, F. R.: 1986, *Internal Combustion Engines*, 1st edn, John Wiley and Sons.
- Filzmoser, P.: 2004, A multivariate outlier detection method, *Seventh International Conference on Computer Data Analysis and Modeling*, Vol. 1, Minsk, Belarus, pp. 18–22.
- Fleming, P. J., Purshouse, R. C. and Lygoe, R. J.: 2005, Many-Objective Optimization: An Engineering Design Perspective, in C. A. C. Coello (ed.), *Proceedings of the Third International Conference on Evolutionary Multi-Criterion Optimization (EMO 2005)*, Vol. 3410 of *Lecture Notes in Computer Science*, Springer-Verlag, Berlin, Heidelberg, Germany, pp. 14–32.
- Fodor, I. K.: 2002, A survey of dimension reduction techniques, *Technical report*, Center for Applied Scientific Computing, Lawrence Livermore National Laboratory.
- Fonseca, C. M. and Fleming, P. J.: 1998a, Multiobjective Optimization and Multiple Constraint Handling with Evolutionary Algorithms — Part I: A Unified Formulation, *IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans* **28**(1), 26–37.
- Fonseca, C. M. and Fleming, P. J.: 1998b, Multiobjective Optimization and Multiple Constraint Handling with Evolutionary Algorithms — Part II: An Application Example, *IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans* **28**(1), 38–47.
- Fowlkes, W. Y. and Creveling, C. M.: 1995, *Engineering methods for robust product design: using Taguchi methods in technology and product development*, Addison-Wesley.
- Frey, H. and Patil, S.: 2002, Identification and review of sensitivity analysis methods, *Risk Analysis* **22**, 553–578.
- Friedman, J. H. and Tukey, J. W.: 1974, A Projection Pursuit Algorithm for Exploratory Data Analysis, *IEEE Transactions of Computers* **c-23**(9), 881–890.

- Gembicki, F. W.: 1974, *Vector optimization for control with performance and parameter sensitivity indices*, PhD thesis, Case Western Reserve University, Cleveland, Ohio.
- Greenwood, G. W., Hu, X. S. and D'Ambrosio, J. G.: 1997, Fitness functions for multiple objective optimization problems: Combining preferences with Pareto rankings, in R. K. Belew and M. D. Vose (eds), *Foundations of Genetic Algorithms 4*, Morgan Kaufmann, San Mateo, California, pp. 437–455.
- Harman, H. H.: 1967, *Modern Factor Analysis*, 2nd edn, University of Chicago Press.
- Hastie, T. J. and Stuetzle, W.: 1989, Principal Curves, *Journal of American Statistical Association* **84**, 502–516.
- Haug, E. J., Choi, K. K. and Komkov, V.: 1986, *Design Sensitivity Analysis of Structural Systems*, Academic Press, New York.
- Helton, J. C. and Davis, F. J.: 2003, Latin hypercube sampling and the propagation of uncertainty in analyses of complex systems, *Reliability Engineering and System Safety* **81**, 23–69.
- Herzog, P. L., Weissbaeck, M., Herrmuth, H. and CSchuessler, M.: 2007, The Diesel SUV - Pushing Back Emission Frontiers, *Symposium on International Automotive Technology (SIAT 2007)*, The Automotive Research Association of India, Pune, India. SAE Paper No. 2007-26-010.
- Heywood, J. B.: 1988, *Internal Combustion Engine Fundamentals*, McGraw-Hill Series in Mechanical Engineering, McGraw-Hill, Singapore.
- Hiroyasu, T., Miki, M., Kim, M., Watanabe, S., Hiroyasu, H. and Miao, H.: 2004, Reduction of heavy duty diesel engine emission and fuel economy with multi-objective genetic algorithm and phenomenological model, *2004 SAE World Congress*, SAE International, Detroit, Michigan. SAE paper no.: 2004-01-0531.
- Hiroyasu, T., Miki, M. and Watanabe, S.: 2000, The new model of parallel genetic algorithm in multi-objective optimization problems — divided range multi-objective genetic algorithm, in IEEE Neural Networks Council (ed.),

- Proceedings of the 2000 Congress on Evolutionary Computation (CEC 2000)*, Vol. 1, IEEE Service Center, Piscataway, New Jersey, pp. 333–340.
- Holland, J. H.: 1975, *Adaptation in Natural and Artificial Systems*, The University of Michigan Press, Ann Arbor, Michigan.
- Holliday, T.: 1995, *The Design and Analysis of Engine Mapping Experiments: A Two-Stage Approach*, PhD thesis, University of Birmingham.
- Hotelling, H.: 1936, Analysis of a complex of statistical variables into principal components, *Educational Psychology* (24), 417–441, 498–520.
- Hubert, M., Rousseeuw, P. and Vanden Branden, K.: 2005, ROBPCA: a New Approach to Robust Principal Component Analysis, *Technometrics* **47**, 64–79.
- Hughes, E. J.: 2003, Multiple single objective pareto sampling, *IEEE Congress on Evolutionary Computation, CEC 2003*, IEEE, Canberra, Australia, pp. 2678–2684.
- Hughes, E. J.: 2005, Evolutionary Many-Objective Optimisation: Many Once or One Many?, *IEEE Congress on Evolutionary Computation (CEC'2005)*, Vol. 1, IEEE Service Center, Edinburgh, Scotland, pp. 222–227.
- Hughes, E. J.: 2007, MSOPS-II: A general-purpose Many-Objective optimiser, *Proceedings of 2007 IEEE Congress on Evolutionary Computation - CEC 2007*, IEEE Service Center, Singapore, pp. 3944–3951.
- Hwang, C.-L. and Masud, A. S. M.: 1979, *Multiple Objective Decision Making-Methods and Applications*, Vol. 164 of *Lecture Notes in Economics and Mathematical Systems*, Springer-Verlag, Berlin.
- Hyvärinen, A.: 1999, Survey on Independant Component Analysis, *Neural Computing Surveys* **2**, 94–128.
- Ignizio, J. P.: 1976, *Goal Programming and Extensions*, Lexington Books, D. C. Heath and Company, Lexington.
- Ikeda, K., Kita, H. and Kobayashi, S.: 2001, Failure of Pareto-based MOEAs: Does non-dominated really mean near to optimal?, in IEEE Neural Networks Council (ed.), *Proceedings of the 2001 Congress on Evolutionary Computation (CEC 2001)*, Vol. 2, IEEE Service Center, Piscataway, New Jersey, pp. 957–962.

- Inselberg, A.: 1985, The Plane with Parallel Coordinates, *The Visual Computer* **1**, 69–91.
- Ishibuchi, H., Doi, T. and Nojima, Y.: 2006, Incorporation of scalarizing fitness functions into evolutionary multiobjective optimization algorithms, *Parallel Problem Solving from Nature - PPSN IX*, Vol. 4193 of *Lecture Notes in Computer Science*, Springer, pp. 493–502.
- Ishibuchi, H. and Nojima, Y.: 2007, Optimization of scalarizing functions through evolutionary multiobjective optimization, *Evolutionary Multi-Criterion Optimization - EMO 2007*, Vol. 4403 of *Lecture Notes in Computer Science*, Springer, pp. 51–65.
- Ishibuchi, H., Nojima, Y. and Doi, T.: 2006, Comparison between Single-Objective and Multi-Objective Genetic Algorithms: Performance Comparison and Performance Measures, *IEEE Congress on Evolutionary Computation, 2006 (CEC 2006)*, pp. 1143–1150.
- Ishibuchi, H., Tsukamoto, N. and Nojima, Y.: 2007, Iterative approach to indicator-based multiobjective optimization, *Proceedings of 2007 IEEE Congress on Evolutionary Computation - CEC 2007*, pp. 3967–3974.
- Ishibuchi, H., Tsukamoto, N. and Nojima, Y.: 2008, Behavior of evolutionary many-objective optimization, *Computer Modeling and Simulation, 2008. UKSIM 2008. Tenth International Conference on*, pp. 266–271.
- Jaimes, A. L., Coello, C. A. C. and Chakraborty, D.: 2008, Objective reduction using a feature selection technique, *2008 Genetic and Evolutionary Computation Conference, GECCO 2008*, ACM Press, Atlanta, USA.
- Jain, A. K. and Dubes, R. C.: 1988, *Algorithms for Clustering Data*, Prentice-Hall advanced reference series, Prentice-Hall, Inc., Upper Saddle River, NJ.
- Jain, A. K., Murty, M. N. and Flynn, P. J.: 1999, Data Clustering: A Review, *ACM Computing Surveys* **31**(3).
- Jolliffe, I. T.: 1972, Discarding Variables in a Principal Component Analysis. I: Artificial Data, *Journal of the Royal Statistical Society. Series C (Applied Statistics)* **21**(2), 160–173.

- Jolliffe, I. T.: 1973, Discarding Variables in a Principal Component Analysis. II: Real Data, *Journal of the Royal Statistical Society. Series C (Applied Statistics)* **22**(1), 21–31.
- Jolliffe, I. T.: 2002, *Principal Component Analysis*, 2nd edn, Springer, New York.
- Kaiser, H. F.: 1958, The varimax criterion for analytic rotation in factor analysis, *Psychometrika* **23**, 187–200.
- Kaiser, H. F.: 1960, The application of electronic computers to factor analysis, *Educ. Psychol. Meas.* **20**, 141–151.
- Kambhatla, N. and Leen, T. K.: 1997, Dimension reduction by local principal component analysis, *Neural Computation* **9**(7), 1493–1516.
- Kawaguchi, A., Aiba, T., Takada, N. and Ona, K.: 2009, A robustness-focused shape optimization method for intake ports, *SAE International Powertrains, Fuels and Lubricants Meeting*, SAE International, Florence, Italy. SAE paper no.: 2009-01-1777.
- Kendall, M.: 1975, *Multivariate Analysis*, Charles Griffin & Co.
- Khare, V., Yao, X. and Deb, K.: 2003, Performance Scaling of Multi-Objective Evolutionary Algorithms, in C. M. Fonseca, P. J. Fleming, E. Zitzler, K. Deb and L. Thiele (eds), *Proceedings of the Second International Conference on Evolutionary Multi-Criterion Optimization (EMO 2003)*, Vol. 2632 of *Lecture Notes in Computer Science*, Springer, Berlin, pp. 376–390.
- Knowles, J.: 2009, Closed-loop evolutionary multiobjective optimization, *Computational Intelligence Magazine, IEEE* **4**(3), 77–91.
- Kohonen, T.: 1995, *Self-Organizing Maps*, Springer, Berlin.
- Köppen, M. and Yoshida, K.: 2007, Substitute distance assignments in nsga-ii for handling many-objective optimization problems, *Evolutionary Multi-Criterion Optimization - EMO 2007*, Vol. 4403 of *Lecture Notes in Computer Science*, Springer, Berlin, pp. 727–741.

- Kukkonen, S. and Deb, K.: 2006, A fast and effective method for pruning of non-dominated solutions in many-objective problems, *Parallel Problem Solving from Nature - PPSN IX*, pp. 553–562.
- Kukkonen, S. and Lampinen, J.: 2007, Ranking-dominance and many-objective optimization, *Proceedings of IEEE Congress on Evolutionary Computation (CEC 2007)*, pp. 3983–3990.
- Kume, T., Iwamoto, Y., Iida, K., Murakami, M., Akishino, K. and Ando, H.: 1996, Combustion control technologies for direct injection SI engines, *Proceedings of the SAE International Congress and Exposition*, number 960600 in *SAE Technical Paper*, SAE, Warrendale, PA, USA.
- Kursawe, F.: 1991, A variant of evolution strategies for vector optimization, in H.-P. Schwefel and R. Männer (eds), *Proceedings of the Parallel Problem Solving from Nature I Conference*, Springer-Verlag, Berlin, pp. 193–197.
- Laumanns, M., Thiele, L., Zitzler, E. and Deb, K.: 2002, Archiving with Guaranteed Convergence and Diversity in Multi-Objective Optimization, in E. Cant-Paz, K. Mathias, R. Roy, D. Davis, R. Poli, K. Balakrishnan, V. Honavar, G. Rudolph, J. Wegener, L. Bull, M. A. Potter, A. C. Schultz, J. F. Miller, B. E. and N. Jonoska (eds), *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO'2002)*, Morgan Kaufmann Publishers, pp. 439–447.
- Lee, S., McGee, J. M., Quay, B. D. and Santavicca, D. A.: 2000, Engine cold-start testing, *Automotive Engineering International* pp. 162–166.
- Liu, G.: 1968, *Introduction to Combinatorial Mathematics*, McGraw-Hill, New York.
- Macqueen, J.: 1967, Some methods for classification and analysis of multivariate observations, *5th Berkeley Symposium on Mathematical Statistics and Probability*, Vol. 1, University of California Press, Berkeley, CA, pp. 281–2.
- MathWorks: 2008a, *Model-Based Calibration Toolbox<sup>TM</sup>: Model Browser User's Guide*, The MathWorks, Inc.
- MathWorks: 2008b, moga.m in Ford Add-ons v3.5.0 library to Matlab<sup>®</sup> R2008b MBC<sup>TM</sup> Toolbox.



- McLachlan, G. and Krishnan, T.: 1997, *The EM Algorithm and Extensions*, Wiley, New York.
- Miettinen, K.: 1999, *Nonlinear Multiobjective Optimization*, Kluwer Academic Publishers, Boston.
- Miettinen, K.: 2001, Some methods for nonlinear multi-objective optimization, in E. Zitzler *et al.* (eds), *Proceedings of the First International Conference on Evolutionary Multi-Criterion Optimization (EMO 2001)*, Vol. 1993 of *Lecture Notes in Computer Science*, Springer-Verlag, Berlin, Heidelberg, Germany, pp. 1–20.
- Miettinen, K.: 2002, Interactive nonlinear multiobjective procedures, *Multiple Criteria Optimization: State of the Art Annotated Bibliographic Surveys*, Kluwer Academic Publishers, pp. 227–276.
- Miettinen, K. and Mäkelä, M. M.: 1995, Interactive bundle-based method for nondifferentiable multiobjective optimization: NIMBUS, *Optimization* **34**, 231–246.
- Monarchi, D. E., Kisiel, C. C. and Duckstein, L.: 1973, Interactive multiobjective programming in water resources: A case study, *Water Resources Research* **9**(4), 837–850.
- Montgomery, D. C.: 1991, *Design and Analysis of Experiments*, 3rd edn, John Wiley and Sons.
- Montgomery, D. C. and Peck, E. A.: 1991, *Introduction to linear regression analysis*, John Wiley & Sons.
- Montgomery, D. T. and Reitz, R. D.: 1996, Six-mode cycle evaluation of the effect of egr and multiple injections on particulate and nox emissions from a d.i. diesel engine, *1996 SAE International Congress and Exposition*, SAE Technical Paper Series, Society of Automotive Engineers, Inc., Warrendale, Pennsylvania, USA, Detroit, Michigan, USA. Paper no.: 960316.
- Morrison, A., Ross, G. and Chalmers, M.: 2003, Fast Multidimensional Scaling through Sampling, Springs and Interpolation, *Information Visualization* **2**(1), 68–77.

- Müller, H., Biermann, D., Kersting, P., Michelitsch, T., Begau, C., Heuel, C., Joliet, R., Kolanski, J., Krller, M., Moritz, C., Niggemann, D., Stber, M., Stnner, T., Varwig, J. and Zhai, D.: 2008, Intuitive Visualization and Interactive Analysis of Pareto Sets Applied on Production Engineering System, *in* A. Yang, Y. Shan and L. T. Bui (eds), *Success in Evolutionary Computation*, Vol. 92 of *Studies in Computational Intelligence*, Springer-Verlag, pp. 189–214.
- Mundorff, F., Carstensen, H. and Bierbaumer, J.: 1998, Direct injection internal combustion engines - the automotive industry's contribution to reduced CO<sub>2</sub> emissions, *Proceedings of the Total Life Cycle Conference and Exposition*, number 982217 in *SAE Technical Paper*, SAE, Warrendale, PA, USA.
- Nair, V. N.: 1992, Taguchi's parameter design: A panel discussion, *Technometrics* **34**(2), 127–161.
- Obayashi, S., Jeong, S. and Chiba, K.: 2005, Multi-Objective Design Exploration for Aerodynamic Configurations, *35th AIAA Fluid Dynamics Conference and Exhibit*, Toronto, Canada.
- Obayashi, S. and Sasaki, D.: 2003, Visualization and Data Mining of Pareto Solutions Using Self-Organizing Map, *in* C. M. Fonseca, P. J. Fleming, E. Zitzler, K. Deb and L. Thiele (eds), *Proceedings of the Second International Conference on Evolutionary Multi-Criterion Optimization (EMO 2003)*, Springer, Berlin, pp. 796–809.
- Press, W. H., Flannery, B. P., Teukolsky, S. A. and Vetterling, W. T.: 1992, *Numerical Recipes in C*, Cambridge University Press.
- Preussner, C., Doring, C., Fehler, S. and Kampmann, S.: 1998, Interaction between mixture preparation, combustion system and injector performance, *Proceedings of the SAE International Congress and Exposition*, number 980498 in *Direct Injection SI Engine Technology (SP-1314)*, SAE, Warrendale, PA, USA.
- Purshouse, R. C.: 2003, *On the Evolutionary Optimisation of Many Objectives*, PhD thesis, University of Sheffield, Sheffield, UK.
- Purshouse, R. C. and Fleming, P. J.: 2003a, Conflict, Harmony, and Independence: Relationships in Evolutionary Multi-Criterion Optimisation, *in* C. M. Fonseca, P. J.

- Fleming, E. Zitzler, K. Deb and L. Thiele (eds), *Proceedings of the Second International Conference on Evolutionary Multi-Criterion Optimization (EMO 2003)*, Springer, Berlin, pp. 16–30.
- Purshouse, R. C. and Fleming, P. J.: 2003b, Evolutionary multi-objective optimisation: An exploratory analysis, *2003 Congress on Evolutionary Computation, CEC 2003*, Vol. 3, IEEE Press, Canberra, Australia, pp. 2066–2073.
- Rachmawati, L. and Srinivasan, D.: 2006, Preference incorporation in multi-objective evolutionary algorithms: A survey, *2006 IEEE Congress on Evolutionary Computation, (CEC 2006)*, Vancouver, BC, Canada, pp. 962–968.
- Rao, S. S.: 1996, *Engineering Optimization: Theory and Practice*, 3 edn, Wiley-Interscience.
- Ravindran, A., Ragsdell, K. M. and Reklaitis, G. V.: 2006, *Engineering Optimization: Methods and Applications*, 2nd edn, John Wiley & Sons, Hoboken, New Jersey.
- Rechenberg, I.: 1973, *Evolutionßstrategie: Optimierung technischer systeme nach prinzipien der biologischen evolution*, Frommann-Holzboog, Stuttgart.
- Rousseeuw, P.: 1984, Least Median of Squares Regression, *Journal of the American Statistical Association* **79**, 871–880.
- Rousseeuw, P. J.: 1985, Multivariate Estimation with High Breakdown Point, in I. V. W. Grossman, G. Pflug and W. Wertz (eds), *Mathematical Statistics and Applications*, Vol. B, Reidel, Dordrecht, The Netherlands, pp. 283–297.
- Rousseeuw, P. J.: 1997, Introduction to Positive-Breakdown Methods, in G. S. Maddala and C. R. Rao (eds), *Handbook of Statistics*, Vol. 15, Elsevier, Amsterdam, pp. 101–121.
- Rousseeuw, P. J. and Leroy, A. M.: 1987, *Robust Regression and Outlier Detection*, John Wiley & Sons.
- Rousseeuw, P. and Van Driessen, K.: 1999, A Fast Algorithm for the Minimum Covariance Determinant Estimator, *Technometrics* **41**, 212–223.
- Saltelli, A., Tarantola, S. and Chan, K. P.-S.: 1999, A quantitative model-independent method for global sensitivity analysis of model output, *Technometrics* **41**, 39–56.

- Sasaki, S., Sawada, D., Takanori, U. and Sami, H.: 1998, Effects of EGR on direct injection gasoline engine, *Japanese Society of Automotive Engineers Review*, Vol. 19, JSAE, pp. 223–228.
- Sato, H., Aguirre, H. and Tanaka, K.: 2007, Controlling dominance area of solutions and its impact on the performance of moeas, *Evolutionary Multi-Criterion Optimization - EMO 2007*, Vol. 4403 of *Lecture Notes in Computer Science*, Springer, Berlin, pp. 5–20.
- Savitzky, A. and Golay, M. J. E.: 1964, Smoothing and differentiation of data by simplified least-squares procedures, *Analytical Chemistry* **36**, 1627–1639.
- Saxena, D. K. and Deb, K.: 2007, Non-linear dimensionality reduction procedures for certain large-dimensional multi-objective optimization problems: Employing correntropy and a novel maximum variance unfolding, *Proceedings of the Fourth International Conference on Evolutionary Multi-Criterion Optimization (EMO 2007)*.
- Schaffer, J. D.: 1985, Multiple objective optimization with vector evaluated genetic algorithms, in J. Grefenstette (ed.), *Proceedings of the First International Conference on Genetic Algorithms*, Lawrence Erlbaum Associates, Mahwah, New Jersey, pp. 93–100.
- Schroder, P.: 1998, *Multivariable Control of Active Magnetic Bearings*, PhD thesis, University of Sheffield, Sheffield, UK.
- Searle, S. R.: 1982, *Matrix Algebra Useful for Statistics*, 1st edn, John Wiley and Sons.
- Shayler, P. J., Darnton, N. J. and Ma, T.: 1996, The prediction of fuel consumption and emissions during warm-up, *IMEchE International Seminar on Application of Powertrain and Fuel Technologies to Meet Emissions Standards*, Mechanical Engineering Publications, London, pp. 53–64. paper C517/009/96.
- Shenfield, A., Fleming, P. J. and Alkarouri, M.: 2007, Computational steering of a multi-objective evolutionary algorithm for engineering design, *Engineering Applications of Artificial Intelligence* **20**(8), 1047–1057.

- Srinivas, N. and Deb, K.: 1994, Multiobjective optimization using nondominated sorting in genetic algorithms, *Evolutionary Computation* **2**(3), 221–248.
- Stevens, J.: 1986, *Applied multivariate statistics for the social sciences*, Lawrence Erlbaum Associates, Hillsdale, NJ.
- Stone, R.: 1994, *Introduction to Internal Combustion Engines*, 2nd edn, Macmillan Press.
- Streichert, F., Ulmer, H. and Zell, A.: 2005, Parallelization of multi-objective evolutionary algorithms using clustering algorithms, *Conference on Evolutionary Multi-Criterion Optimization, EMO 2005*, Vol. 3410 of *Lecture Notes in Computer Science*, Springer-Verlag, Guanajuato, Mexico, pp. 92–107.
- Sülflow, A., Drechsler, N. and Drechsler, R.: 2007, Robust multi-objective optimization in high dimensional spaces, *Evolutionary Multi-Criterion Optimization - EMO 2007*, Vol. 4403 of *Lecture Notes in Computer Science*, pp. 715–726.
- Suppakitnarm, A., Seffen, K. A., Parks, G. T., Clarkson, P. J. and Liu, J. S.: 1999, Design by multiobjective optimization using simulated annealing, *Proceedings of the International Conference on Engineering Design (ICED 99)*.
- Taguchi, G.: 1987, *System of Experimental Design: Engineering Methods to Optimize Quality and Minimize Cost*, UNIPUB/Kraus International, White Plains, New York.
- Taguchi, G.: 1993, *Taguchi on Robust Technology Development: Bringing Quality Engineering Upstream*, ASME Press.
- Tan, K. C., Khor, E. F., Lee, T. H. and Sathikannan, R.: 2003, An Evolutionary Algorithm With Advanced Goal and Priority Specification for Multi-Objective Optimization, *Journal of Artificial Intelligence Research* **18**, 183–215.
- Tang, Y., Reed, P., Wagener, T. and van Werkoven, K.: 2007, Comparing sensitivity analysis methods to advance lumped watershed model identification and evaluation, *Hydrology and Earth System Sciences* **11**, 793–817.
- Wagner, T., Beume, N. and Naujoks, B.: 2007, Pareto-, aggregation-, and indicator-based methods in many-objective optimization, *Evolutionary*

- Multi-Criterion Optimization - EMO 2007*, Vol. 4403 of *Lecture Notes in Computer Science*, Springer, Berlin, pp. 742–756.
- Wegman, E. J.: 1990, Hyperdimensional Data Analysis Using Parallel Coordinates, *Journal of the American Statistical Association* **85**, 664–675.
- Wegman, E. J. and Luo, Q.: 1997, High Dimensional Clustering Using Parallel Coordinates and the Grand Tour, *Computing Science and Statistics* **28**, 352–360.
- Wiemer, S., Kubach, H. and Spicher, U.: 2007, Investigations on the start-up process of a disi engine, *Powertrain and Fluid Systems Conference and Exhibition*, SAE Technical Paper Series, SAE International, Rosemont, Illinois. SAE paper no.: 2007-01-4012.
- Xu, L., Krzyzak, A. and Oja, E.: 1993, Rival Penalized Competitive Learning for Clustering Analysis, RBF Net, and Curve Detection, *IEEE Transactions on Neural Networks* **4**(4).
- Xu, R. and Wunsch, D.: 2005, Survey of Clustering Algorithms, *IEEE Transactions of Neural Networks* **16**(3).
- Yi, J., Han, Z., Yang, J., Anderson, R., Trigui, N. and Boussarsar, R.: 2000, Modeling of the interaction of intake flow and fuel spray in DISI engines, *Proceedings of the SAE 2000 World Congress*, number 2000-01-0656 in *Multi-Dimensional Engine Modeling (SP1512)*, SAE, Warrendale, PA, USA.
- Yoshikawa, T., Yamashiro, D. and Furuhashi, T.: 2007, A Proposal of Visualization of Multi-Objective Pareto Solutions - Development of Mining Technique for Solutions, *IEEE Symposium on Computational Intelligence in Multicriteria Decision Making (MCDM'2007)*, IEEE Press, Honolulu, Hawaii, USA, pp. 172–177.
- Yu, P. L.: 1973, A class of solutions for group decision problems, *Management Science* **19**, 936–946.
- Yun, H. and Reitz, R. D.: 2003, An experimental study on emissions optimization using micro-genetic algorithms in a hsd diesel engine, *SAE World Congress*, SAE Technical Paper Series, SAE International, Detroit, Michigan. SAE paper no.: 2003-01-0347.

- Zahn, C. T.: 1971, Graph-theoretical methods for detecting and describing gestalt clusters, *IEEE Transactions on Computers* **C-20**, 68–86.
- Zeleny, M.: 1973, Compromise programming, in J. L. Cochrane and M. Zeleny (eds), *Multiple Criteria Decision Making*, University of South Carolina Press, Columbia, South Carolina, pp. 262–301.
- Zhao, F., Lai, M. C. and Harrington, D. L.: 1999, Automotive spark-ignition direct-injection gasoline engines, *Progress in Energy and Combustion Science* **25**, 437–562.
- Zitzler, E., Deb, K. and Thiele, L.: 2000, Comparison of multiobjective evolutionary algorithms: Empirical results, *Evolutionary Computation* **8**(2), 173–195.
- Zitzler, E. and Künzli, S.: 2004, Indicator-based selection in multiobjective search, in X. Y. et al (ed.), *Parallel Problem Solving from Nature - PPSN VIII*, Vol. 3242 of *Lecture Notes in Computer Science*, Springer-Verlag, Birmingham, UK, pp. 832–842.
- Zitzler, E., Laumanns, M. and Thiele, L.: 2001, SPEA2: Improving the strength Pareto evolutionary algorithm, *TIK Report 103*, Swiss Federal Institute of Technology (ETH), Zurich, Switzerland.
- Zitzler, E. and Thiele, L.: 1999, Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach, *IEEE Transactions on Evolutionary Computation* **3**(4), 257–271.
- Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C. M. and Grunert da Fonseca, V.: 2003, Performance assessment of multiobjective optimizers: An analysis and review, *IEEE Transactions on Evolutionary Computation* **7**(2), 117–132.
- Zou, X., Chen, Y., Liu, M. and Kang, L.: 2008, A new evolutionary algorithm for solving many-objective optimization problems, *Systems, Man, and Cybernetics, Part B, IEEE Transactions on* **38**(5), 1402–1412.
- Zou, X., Liu, M., Kang, L. and He, J.: 2004, A high performance multi-objective evolutionary algorithm based on the principles of thermodynamics, *Parallel Problem Solving from Nature - PPSN VIII*, pp. 922–931.