# DISSERTATION

# USE OF MULTI-OBJECTIVE PARTICLE SWARM OPTIMIZATION IN WATER RESOURCES MANAGEMENT

Submitted by

Alexandre Moreira Baltar

Department of Civil and Environmental Engineering

In partial fulfillment of the requirements

for the Degree of Doctor of Philosophy

Colorado State University

Fort Collins, Colorado

Summer 2007

COLORADO STATE UNIVERSITY

April 2, 2007

WE HEREBY RECOMMEND THAT THE DISSERTATION PREPARED UNDER OUR SUPERVISION BY ALEXANDRE MOREIRA BALTAR ENTITLED USE OF MULTI-OBJECTIVE PARTICLE SWARM OPTIMIZATION IN WATER RESOURCES MANAGEMENT BE ACCEPTED AS FULFILLING IN PART REQUIREMENTS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY.

Committee on Graduate Work

*(Please print name under signature)*

Neil S. Grigg

Luiz Gabriel T. Azevedo

John B. Loomis

DARRELL G. FONTANE

**Adviser**

**Department Head**

**ABSTRACT OF DISSERTATION**

USE OF MULTI-OBJECTIVE PARTICLE SWARM OPTIMIZATION IN WATER RESOURCES MANAGEMENT

Water resources management presents a large variety of multi-objective problems that require powerful optimization tools to fully characterize the existing trade-offs. Different optimization methods, based on mathematical programming at first and on evolutionary algorithms (EA) more recently, have been applied with various degrees of success. This dissertation presents a multi-objective implementation of a relatively recent evolutionary technique called particle swarm optimization (PSO). The multi-objective PSO (MOPSO) algorithm was implemented as a generalized solver for Microsoft Excel®, and applied to a set of test functions commonly used in the EA literature and to selected water resources management problems, including a classic multi-purpose reservoir operation problem, a problem of selective withdrawal from thermally stratified reservoirs, and a reservoir operation problem using storage guide curves with fuzzy objectives.

Three other multi-objective solvers were developed: a second EA approach, using the non-dominated sorting genetic algorithm II (NSGA-II), a traditional mathematical programming method, the ε-constraint with nonlinear optimization, and a pure random search approach.

In most problems, the MOPSO and the NSGA-II algorithms provided good approximations to the true Pareto optimal sets. The NSGA-II algorithm seems to be more robust perfoming well in a wider range of problems, although MOPSO showed better performance for some problems. The main advantage of the MOPSO is associated with the simplicity of the algorithm. The basic MOPSO algorithm is much simpler and easier to implement than the NSGA-II. This makes MOPSO more flexible to accommodate necessary changes to deal with specific problems.

A method to visualize and explore solutions of multi-objective optimization was introduced. The Interactive Compromise Coordinate (ICC) method allows the projection of all alternative solutions in a single unit circle graph. The decision maker (DM) can explore Pareto optimal sets and rank alternatives using a compromise programming approach based on weights that can be interactively changed. The method's basic assumptions are that the DM's preference structure can be modeled by a set of weights and that all alternatives are transitively comparable to each other, i.e. a complete pre-order is obtainable. The mathematical basis for the method is presented and the method of projection is illustrated.

Alexandre Moreira Baltar
Department of Civil and Environmental Engineering
Colorado State University
Fort Collins, CO 80523
Summer 2007

## ACKNOWLEDGEMENTS

The author's adviser, Dr. Darrell G. Fontane, is most thankfully acknowledged for his support, academic and personal, his guidance through these years, and his friendship.

The author would like to thank the other members of the graduate committee at Colorado State University, Dr. Neil S. Grigg, Department of Civil and Environmental Engineering, Dr. John B. Loomis, Department of Agricultural and Resource Economics, and Dr. Luiz Gabriel T. Azevedo, the World Bank, for the encouragement and the constructive supervision of this research.

The author is indebted to the Department of Civil and Environmental Engineering, Colorado State University, to all its faculty and staff, and particularly to Dr. José D. Salas and Dr. John W. Labadie, for their academic support and challenging courses, and to Marilee Rowe, for her admirable and continuous willingness to help.

The author would like to acknowledge the support of the CNPq, a Brazilian Government agency dedicated to the development of science and technology, which has funded his PhD studies at Colorado State University.

Finally, the author thanks all his family for the invaluable support through these years, and particularly his mother-in-law, for her loving and unconditional help in Fort Collins and in Brazil.

## DEDICATION

To my wife, Eike, and my sons, Pedro and André

## TABLE OF CONTENTS

ABSTRACT OF DISSERTATION iii
ACKNOWLEDGEMENTS v
DEDICATION vi
LIST OF TABLES x
LIST OF FIGURES xv
LIST OF ACRONYMS xix

**1. INTRODUCTION 1**

**2. LITERATURE REVIEW 7**

**2.2. Mathematical Programming** 9
2.2.1. Generating Methods 9
*The Weighting Method* 10
*The ε-Constraint Method* 11
2.2.2. Methods with Prior Articulation of Preferences 12
*Goal Programming* 13
*PROMETHEE II* 14
*ELECTRE III* 16
*Compromise Programming* 18

**2.3. Evolutionary Multi-Objective Optimization in Water Resources** 20

**2.4. Multi-Objective Particle Swarm Optimization** 23

**3. RESEARCH STEPS, TEST PROBLEMS AND COMPARISON METRICS 29**

**3.1. Test Functions** 30

**3.2. Selected Water Resources Management Problems** 32

**3.3. Comparison Metrics** 32
3.3.1. Generational Distance (GD) 33
3.3.2. Spacing Metric (SP) 34
3.3.3. Diversity Metric (DI) 35
3.3.4. Inverted Generational Distance (iGD) 35
3.3.5. Dominated Ratio (DR) 37
3.3.6. Dominated Degree (DD) 38

**4. THE METHODS: DESCRIPTION AND IMPLEMENTATION** **40**

**4.1. Multi-Objective Particle Swarm Optimization** 43

4.1.1. MOPSO Solver Specific Features 45

*Scripting Tool for User-Defined Macros* 45

*Routine for Handling Equality Constraints Imposed on Decision Variables* 46

*MOPSO Specific Operator for Extreme Solutions* 50

4.1.2. MOPSO Parameters 53

**4.2. Non-Dominated Sorting Genetic Algorithm II** 54

4.2.1. NSGA-II Solver Specific Features 56

4.2.2. NSGA-II Parameters 57

**4.3. ε-Constraint with Nonlinear Optimization** 58

4.3.1. ε-NLP Solver Specific Features 59

4.3.2. ε-NLP Solver Parameters 61

**4.4. Random Search Solver** 62

**5. THE INTERACTIVE COMPROMISE COORDINATE METHOD** **64**

**5.1. Methods for Displaying Results of Multi-Objective Analyses** 65

**5.2. The Interactive Compromise Coordinate Method** 67

**6. APPLICATIONS** **73**

**6.1. Test Functions** 74

6.1.1. Test Problem 1 74

*Analysis and Comparison* 77

6.1.2. Test Problem 2 78

*Analysis and Comparison* 83

6.1.3. Test Problem 3 85

*Analysis and Comparison* 88

6.1.4. Test Problem 4 90

*Analysis and Comparison* 93

6.1.5. Test Problem 5 95

*Analysis and Comparison* 98

6.1.6. Analysis of Results for the Test Functions 100

**6.2. Selected Water Resources Management Problems** 101

6.2.1. Multi-Purpose Reservoir Operational Planning 101

*Test Problem WE* 103

Analysis and Comparison 106

Best Compromise Solutions 106

*Test Problem WEF* 108

Analysis and Comparison 112

Best Compromise Solutions 114

*Test Problem WEFR* 115

Analysis and Comparison 119
Best Compromise Solutions 121
6.2.2. Selective Withdrawal from a Thermally Stratified Reservoir 122
*Test Problem TD* 125
Analysis and Comparison 128
Best Compromise Solutions 130
*Test Problem TDT* 131
Analysis and Comparison 136
Best Compromise Solutions 138
*Test Problem TDT2* 140
Analysis and Comparison 146
Best Compromise Solutions 148
6.2.3. Multi-Purpose Rule-Based Reservoir Operation with Fuzzy Objectives 149
*Test Problem FUZZY* 152
Analysis and Comparison 154
Best Compromise Solutions 155
6.2.4. Analysis of Results for Water Resources Management Problems 158

**6.3. Sensitivity Analysis of the Parameters** 160

**7. CONCLUSIONS AND RECOMMENDATIONS 163**

**REFERENCES 169**

**APPENDIX A: STATISTICS OF PERFORMANCE METRICS 177**

**APPENDIX B: GRAPHS OF THE SENSITIVITY ANALYSIS 229**

**APPENDIX C: INTERFACES OF THE IMPLEMENTED SOLVERS 240**

## LIST OF TABLES

Table 2.1 – Applications of the weighting method in water resources ..... 11
Table 2.2 – Applications of the ε-constraint method in water resources ..... 12
Table 2.3 – Applications of the goal programming method in water resources ..... 14
Table 2.4 – Applications of the PROMETHEE, ELECTRE and CP in water resources . 20
Table 2.5 – Applications of multi-objective evolutionary algorithms in water resources 23
Table 2.6 – Characteristics of selected multi-objective PSO approaches ..... 28

Table 6.1 – Generational Distance Metric for Test Problem 1 ..... 75
Table 6.2 – GD Reference Values for Test Problem 1 ..... 75
Table 6.3 – Spacing Metric for Test Problem 1 ..... 76
Table 6.4 – Spacing Reference Values for Test Problem 1 ..... 76
Table 6.5 – Inverted Generational Distance Metric for Test Problem 1 ..... 76
Table 6.6 – Diversity Metric for Test Problem 1 ..... 76
Table 6.7 – Processing Time for Test Problem 1 ..... 77
Table 6.8 – Dominated Ratio for Test Problem 1 ..... 77
Table 6.9 – Dominated Degree for Test Problem 1 ..... 77
Table 6.10 – Generational Distance Metric for Test Problem 2 ..... 80
Table 6.11 – Generational Distance Reference Values for Test Problem 2 ..... 81
Table 6.12 – Spacing Metric for Test Problem 2 ..... 81
Table 6.13 – Spacing Reference Values for Test Problem 2 ..... 81
Table 6.14 – Inverted Generational Distance Metric for Test Problem 2 ..... 81
Table 6.15 – Diversity Metric for Test Problem 2 ..... 82
Table 6.16 – Processing Time for Test Problem 2 ..... 82
Table 6.17 – Dominated Ratio for Test Problem 2 ..... 82
Table 6.18 – Dominated Degree for Test Problem 2 ..... 82
Table 6.19 – Generational Distance Metric for Test Problem 3 ..... 86

Table 6.20 – Generational Distance Reference Values for Test Problem 3 .................... 86
Table 6.21 – Spacing Metric for Test Problem 3 .................... 86
Table 6.22 – Spacing Reference Values for Test Problem 3 .................... 87
Table 6.23 – Inverted Generational Distance Metric for Test Problem 3 .................... 87
Table 6.24 – Diversity Metric for Test Problem 3 .................... 87
Table 6.25 – Diversity Reference Values for Test Problem 3 .................... 87
Table 6.26 – Processing Time for Test Problem 3 .................... 88
Table 6.27 – Dominated Ratio for Test Problem 3 .................... 88
Table 6.28 – Dominated Degree for Test Problem 3 .................... 88
Table 6.29 – Generational Distance Metric for Test Problem 4 .................... 91
Table 6.30 – Generational Distance Reference Values for Test Problem 4 .................... 91
Table 6.31 – Spacing Metric for Test Problem 4 .................... 92
Table 6.32 – Inverted Generational Distance Metric for Test Problem 4 .................... 92
Table 6.33 – Diversity Metric for Test Problem 4 .................... 92
Table 6.34 – Diversity Reference Values for Test Problem 4 .................... 92
Table 6.35 – Processing Time for Test Problem 4 .................... 93
Table 6.36 – Dominated Ratio for Test Problem 4 .................... 93
Table 6.37 – Dominated Degree for Test Problem 4 .................... 93
Table 6.38 – Generational Distance Metric for Test Problem 5 .................... 96
Table 6.39 – Spacing Metric for Test Problem 5 .................... 97
Table 6.40 – Inverted Generational Distance Metric for Test Problem 5 .................... 97
Table 6.41 – Diversity Metric for Test Problem 5 .................... 97
Table 6.42 – Processing Time for Test Problem 5 .................... 97
Table 6.43 – Dominated Ratio for Test Problem 5 .................... 98
Table 6.44 – Dominated Degree for Test Problem 5 .................... 98
Table 6.45 – Number of times each method showed the best average metric .................... 101
Table 6.46 – Generational Distance Metric for Test Problem WE .................... 104
Table 6.47 – Spacing Metric for Test Problem WE .................... 104
Table 6.48 – Inverted Generational Distance Metric for Test Problem WE .................... 105
Table 6.49 – Diversity Metric for Test Problem WE .................... 105
Table 6.50 – Processing Time for Test Problem WE .................... 105
Table 6.51 – Dominated Ratio for Test Problem WE .................... 105
Table 6.52 – Dominated Degree for Test Problem WE .................... 106

Table 6.53 – Best Compromise Solutions for Test Problem WE .................................. 108
Table 6.54 – Generational Distance Metric for Test Problem WEF.............................. 110
Table 6.55 – Spacing Metric for Test Problem WEF .................................................. 110
Table 6.56 – Inverted Generational Distance Metric for Test Problem WEF ................ 111
Table 6.57 – Processing Time for Test Problem WEF ................................................. 111
Table 6.58 – Dominated Ratio for Test Problem WEF ................................................. 111
Table 6.59 – Dominated Degree for Test Problem WEF.............................................. 111
Table 6.60 – Best Compromise Solutions for Test Problem WEF ................................ 114
Table 6.61 – Generational Distance Metric for Test Problem WEFR........................... 117
Table 6.62 – Spacing Metric for Test Problem WEFR................................................. 118
Table 6.63 – Inverted Generational Distance Metric for Test Problem WEFR............. 118
Table 6.64 – Processing Time for Test Problem WEFR .............................................. 118
Table 6.65 – Dominated Ratio for Test Problem WEFR.............................................. 118
Table 6.66 – Dominated Degree for Test Problem WEFR........................................... 119
Table 6.67 – Pair-wise Comparison on Average DR for Problem WEFR .................... 119
Table 6.68 – Pair-wise Comparison on Average DD for Problem WEFR .................... 119
Table 6.69 – Best Compromise Solutions for Test Problem WEFR ............................ 121
Table 6.70 – Generational Distance Metric for Test Problem TD................................ 127
Table 6.71 – Spacing Metric for Test Problem TD ...................................................... 127
Table 6.72 – Inverted Generational Distance Metric for Test Problem TD .................. 127
Table 6.73 – Diversity Metric for Test Problem TD .................................................... 127
Table 6.74 – Processing Time for Test Problem TD .................................................... 128
Table 6.75 – Dominated Ratio for Test Problem TD.................................................... 128
Table 6.76 – Dominated Degree for Test Problem TD................................................. 128
Table 6.77 – Best Compromise Solutions for Test Problem TD .................................. 131
Table 6.78 – Generational Distance Metric for Test Problem TDT .............................. 133
Table 6.79 – Spacing Metric for Test Problem TDT.................................................... 133
Table 6.80 – Inverted Generational Distance Metric for Test Problem TDT................ 134
Table 6.81 – Processing Time for Test Problem TDT.................................................. 134
Table 6.82 – Dominated Ratio for Test Problem TDT ................................................. 134
Table 6.83 – Dominated Degree for Test Problem TDT .............................................. 134
Table 6.84 – Best Compromise Solutions for Test Problem TDT................................ 139
Table 6.85 – Generational Distance Metric for Test Problem TDT2 ........................... 144

Table 6.86 – Spacing Metric for Test Problem TDT2 .......... 144
Table 6.87 – Inverted Generational Distance Metric for Test Problem TDT2 .......... 144
Table 6.88 – Processing Time for Test Problem TDT2 .......... 145
Table 6.89 – Dominated Ratio for Test Problem TDT2 .......... 145
Table 6.90 – Dominated Degree for Test Problem TDT2 .......... 145
Table 6.91 – Pair-wise Comparison on Average DR for Test Problem TDT2 .......... 145
Table 6.92 – Pair-wise Comparison on Average DD for Test Problem TDT2 .......... 146
Table 6.93 – Best Compromise Solutions for Test Problem TDT2 .......... 148
Table 6.94 – Pair-wise Comparison on Average DR for Test Problem FUZZY .......... 154
Table 6.95 – Pair-wise Comparison on Average DD for Test Problem FUZZY .......... 154
Table 6.96 – Best Compromise Solutions for Test Problem FUZZY .......... 156
Table 6.97 – Variations of Parameters for Sensitivity Analysis .......... 161

Table A.1 – MOPSO Statistics for Test Problem 1 .......... 178
Table A.2 – MOPSO-e Statistics for Test Problem 1 .......... 179
Table A.3 – MOPSO-p Statistics for Test Problem 1 .......... 180
Table A.4 – NSGA-II Statistics for Test Problem 1 .......... 181
Table A.5 – RSearch Statistics for Test Problem 1 .......... 182
Table A.6 – MOPSO Statistics for Test Problem 2 .......... 183
Table A.7 – MOPSO-e Statistics for Test Problem 2 .......... 184
Table A.8 – MOPSO-p Statistics for Test Problem 2 .......... 185
Table A.9 – NSGA-II Statistics for Test Problem 2 .......... 186
Table A.10 – RSearch Statistics for Test Problem 2 .......... 187
Table A.11 – MOPSO Statistics for Test Problem 3 .......... 188
Table A.12 – MOPSO-e Statistics for Test Problem 3 .......... 189
Table A.13 – MOPSO-p Statistics for Test Problem 3 .......... 190
Table A.14 – NSGA-II Statistics for Test Problem 3 .......... 191
Table A.15 – RSearch Statistics for Test Problem 3 .......... 192
Table A.16 – MOPSO Statistics for Test Problem 4 .......... 193
Table A.17 – MOPSO-e Statistics for Test Problem 4 .......... 194
Table A.18 – MOPSO-p Statistics for Test Problem 4 .......... 195
Table A.19 – NSGA-II Statistics for Test Problem 4 .......... 196
Table A.20 – RSearch Statistics for Test Problem 4 .......... 197

Table A.21 – MOPSO Statistics for Test Problem 5 .... 198
Table A.22 – MOPSO-e Statistics for Test Problem 5 .... 199
Table A.23 – MOPSO-p Statistics for Test Problem 5 .... 200
Table A.24 – NSGA-II Statistics for Test Problem 5 .... 201
Table A.25 – RSearch Statistics for Test Problem 5 .... 202
Table A.26 – MOPSO Statistics for Test Problem WE .... 203
Table A.27 – MOPSO-e Statistics for Test Problem WE .... 204
Table A.28 – MOPSO-p Statistics for Test Problem WE .... 205
Table A.29 – NSGA-II Statistics for Test Problem WE .... 206
Table A.30 – RSearch Statistics for Test Problem WE .... 207
Table A.31 – MOPSO Statistics for Test Problem WEF .... 208
Table A.32 – MOPSO-e Statistics for Test Problem WEF .... 209
Table A.33 – MOPSO-p Statistics for Test Problem WEF .... 210
Table A.34 – NSGA-II Statistics for Test Problem WEF .... 211
Table A.35 – RSearch Statistics for Test Problem WEF .... 212
Table A.36 – MOPSO Statistics for Test Problem WEFR .... 213
Table A.37 – MOPSO-e Statistics for Test Problem WEFR .... 214
Table A.38 – MOPSO-p Statistics for Test Problem WEFR .... 215
Table A.39 – NSGA-II Statistics for Test Problem WEFR .... 216
Table A.40 – RSearch Statistics for Test Problem WEFR .... 217
Table A.41 – MOPSO Statistics for Test Problem TD .... 218
Table A.42 – MOPSO-e Statistics for Test Problem TD .... 219
Table A.43 – NSGA-II Statistics for Test Problem TD .... 220
Table A.44 – RSearch Statistics for Test Problem TD .... 221
Table A.45 – MOPSO Statistics for Test Problem TDT .... 222
Table A.46 – MOPSO-e Statistics for Test Problem TDT .... 223
Table A.47 – NSGA-II Statistics for Test Problem TDT .... 224
Table A.48 – RSearch Statistics for Test Problem TDT .... 225
Table A.49 – MOPSO Statistics for Test Problem TDT2 .... 226
Table A.50 – NSGA-II Statistics for Test Problem TDT2 .... 227
Table A.51 – RSearch Statistics for Test Problem TDT2 .... 228

## LIST OF FIGURES

Figure 1.1 – Pareto Fronts ..... 3

Figure 2.1 – PROMETHEE Preference Relation Type V ..... 15
Figure 2.2 – PROMETHEE II Algorithm Flowchart ..... 16
Figure 2.3 – ELECTRE III Algorithm Flowchart ..... 18
Figure 2.4 – Compromise Programming Algorithm Flowchart ..... 19
Figure 2.5 – Flowchart of a General EA ..... 21

Figure 3.1 – Flowchart View of the Methodology ..... 31
Figure 3.2 – Computation of the Generational Distance Metric ..... 33
Figure 3.3 – Computation of the Spacing Metric ..... 34
Figure 3.4 – Computation of the Diversity Metric ..... 36
Figure 3.5 – Computation of the Inverted Generational Distance Metric ..... 37
Figure 3.6 – Misleading Information from Performance Metrics ..... 37
Figure 3.7 – Computation of the Dominated Ratio Metric ..... 38
Figure 3.8 – Computation of the Dominated Degree Metric ..... 39
Figure 4.1 – MOPSO Solver Interface ..... 41
Figure 4.2 – ICC Method Imbedded in the MOPSO Solver ..... 42
Figure 4.3 – Flowchart of the MOPSO Solver Algorithm ..... 46
Figure 4.4 – MOPSO Solver Scripting Tool ..... 47
Figure 4.5 – MOPSO Solver Add Constraint Dialog ..... 48
Figure 4.6 – Routine for Handling Equality Constraints Imposed on Decision Variables ..... 49
Figure 4.7 – Sub-Optimal Solutions on the Extreme of Pareto Fronts ..... 50
Figure 4.8 – Pseudo-Code of MOPSO Post-Processing Routine ..... 52
Figure 4.9 – MOPSO Solver Default Parameters ..... 53

Figure 4.10 – Pseudo-Code of NSGA-II Crowding Distance Assignment Procedure ..... 55
Figure 4.11 – NSGA-II Procedure ..... 55
Figure 4.12 – Effect of the Distribution Index on the NSGA-II Crossover Operator ..... 57
Figure 4.13 – NSGA-II Solver Default Parameters ..... 58
Figure 4.14 – Pseudo-Code of the Slope-Based Routine for Epsilon Bounds ..... 60
Figure 4.15 – Effect of the Slope-Based Routine for Epsilon Bounds ..... 61
Figure 4.16 – ε-NLP Solver Default Parameters ..... 62
Figure 5.1 – ICC Input Form ..... 70
Figure 5.2 – ICC Interface ..... 70
Figure 5.3 – ICC Visualization of Known 3-D Surfaces ..... 71
Figure 5.4 – ICC Graph for a 2-D Pareto Curve for Different Weights ..... 72
Figure 6.1 – Feasible Region and Pareto Optimal Front for Test Problem 1 ..... 75
Figure 6.2 – Results for Test Problem 1 ..... 79
Figure 6.3 – Local and Global Pareto Optimal Fronts for Test Problem 2 ..... 80
Figure 6.4 – Results for Test Problem 2 ..... 84
Figure 6.5 – Pareto Optimal Set for Test Problem 3 ..... 85
Figure 6.6 – Results for Test Problem 3 ..... 89
Figure 6.7 – Global and Local Pareto Optimal Fronts for Test Problem 4 ..... 91
Figure 6.8 – Results for Test Problem 4 ..... 94
Figure 6.9 – Feasible Region and Pareto Optimal Front for Problem 5 ..... 96
Figure 6.10 – Results for Test Problem 5 ..... 99
Figure 6.11 – Pareto Optimal Front for Test Problem WE ..... 104
Figure 6.12 – Results for Test Problem WE ..... 107
Figure 6.13 – Best Compromise Solutions for Test Problem WE ..... 108
Figure 6.14 – Pareto Optimal Surface for Test Problem WEF ..... 109
Figure 6.15 – ICC Graph of Pareto Optimal Set for Test Problem WEF ..... 110
Figure 6.16 – Results for Test Problem WEF ..... 113
Figure 6.17 – Best Compromise Solutions for Test Problem WEF ..... 115
Figure 6.18 – Pareto Optimal Set for Test Problem WEFR ..... 117
Figure 6.19 – Results for Test Problem WEFR ..... 120
Figure 6.20 – Best Compromise Solutions for Test Problem WEFR ..... 122
Figure 6.21 – Water Quality Profiles for the SELECT Model ..... 124

Figure 6.22 – Pareto Optimal Front for Test Problem TD ..... 126
Figure 6.23 – Results for Test Problem TD ..... 129
Figure 6.24 – Best Compromise Solutions for Test Problem TD ..... 131
Figure 6.25 – Pareto Optimal Surface for Test Problem TDT ..... 132
Figure 6.26 – ICC Graph of Pareto Optimal Set for Test Problem TDT ..... 133
Figure 6.27 – Results for Test Problem TDT ..... 135
Figure 6.28 – NSGA-II Intermediate Generations for Test Problem TDT ..... 137
Figure 6.29 – Best Compromise Solutions for Test Problem TDT ..... 139
Figure 6.30 – Pareto Optimal Surface for Test Problem TDT2 without Binary Constraints ..... 140
Figure 6.31 – ICC Graph for Test Problem TDT2 without Binary Constraints ..... 141
Figure 6.32 – Pareto Optimal Surface for Test Problem TDT2 ..... 142
Figure 6.33 – ICC Graph of Pareto Optimal Set for Test Problem TDT2 ..... 143
Figure 6.34 – Different Subsets of Solutions for Test Problem TDT2 ..... 143
Figure 6.35 – Results for Test Problem TDT2 ..... 147
Figure 6.36 – Best Compromise Solutions for Test Problem TDT2 ..... 148
Figure 6.37 – Proposed Grey Mountain Reservoir Site ..... 149
Figure 6.38 – ICC Graph of Non-dominated Solutions for Test Problem FUZZY ..... 153
Figure 6.39 – Results for Test Problem FUZZY ..... 155
Figure 6.40 – Best Compromise Solution for Test Problem FUZZY ..... 156
Figure 6.41 – Best Solutions for Each Objective for Test Problem FUZZY ..... 157
Figure 6.42 – Best Guide Curves for Test Problem FUZZY ..... 158

Figure B.1 – Sensitivity Analysis for Parameter $w$ for Test Problem 1 ..... 230
Figure B.2 – Sensitivity Analysis for Parameter $w$ for Test Problem 5 ..... 230
Figure B.3 – Sensitivity Analysis for Parameter $w$ for Test Problem TD ..... 230
Figure B.4 – Sensitivity Analysis for Parameter $w$ for Test Problem WEF ..... 231
Figure B.5 – Sensitivity Analysis for Parameter $c_1$&$c_2$ for Test Problem 1 ..... 231
Figure B.6 – Sensitivity Analysis for Parameter $c_1$&$c_2$ for Test Problem 5 ..... 231
Figure B.7 – Sensitivity Analysis for Parameter $c_1$&$c_2$ for Test Problem TD ..... 232
Figure B.8 – Sensitivity Analysis for Parameter $c_1$&$c_2$ for Test Problem WEF ..... 232
Figure B.9 – Sensitivity Analysis for Parameter *MutRate* for Test Problem 1 ..... 232
Figure B.10 – Sensitivity Analysis for Parameter *MutRate* for Test Problem 5 ..... 233

Figure B.11 – Sensitivity Analysis for Parameter *MutRate* for Test Problem TD ......... 233
Figure B.12 – Sensitivity Analysis for Parameter *MutRate* for Test Problem WEF...... 233
Figure B.13 – Sensitivity Analysis for Parameter $P_{cross}$ for Test Problem 1 .................. 234
Figure B.14 – Sensitivity Analysis for Parameter $P_{cross}$ for Test Problem 5 .................. 234
Figure B.15 – Sensitivity Analysis for Parameter $P_{cross}$ for Test Problem TD............... 234
Figure B.16 – Sensitivity Analysis for Parameter $P_{cross}$ for Test Problem WEF............ 235
Figure B.17 – Sensitivity Analysis for Parameter $\eta_c$ for Test Problem 1....................... 235
Figure B.18 – Sensitivity Analysis for Parameter $\eta_c$ for Test Problem 5....................... 235
Figure B.19 – Sensitivity Analysis for Parameter $\eta_c$ for Test Problem TD .................. 236
Figure B.20 – Sensitivity Analysis for Parameter $\eta_c$ for Test Problem WEF ................ 236
Figure B.21 – Sensitivity Analysis for Parameter $P_{mut}$ for Test Problem 1 .................... 236
Figure B.22 – Sensitivity Analysis for Parameter $P_{mut}$ for Test Problem 5 .................... 237
Figure B.23 – Sensitivity Analysis for Parameter $P_{mut}$ for Test Problem TD................. 237
Figure B.24 – Sensitivity Analysis for Parameter $P_{mut}$ for Test Problem WEF ............. 237
Figure B.25 – Sensitivity Analysis for Parameter $\eta_m$ for Test Problem 1 ...................... 238
Figure B.26 – Sensitivity Analysis for Parameter $\eta_m$ for Test Problem 5 ...................... 238
Figure B.27 – Sensitivity Analysis for Parameter $\eta_m$ for Test Problem TD................... 238
Figure B.28 – Sensitivity Analysis for Parameter $\eta_m$ for Test Problem WEF ............... 239

Figure C.1 – Microsoft Excel Tools Menu with the Links to the Various Solvers ........ 241
Figure C.2 – NSGA-II Solver Interface ........................................................................ 242
Figure C.3 – ε-NLP Solver Interface ............................................................................ 243
Figure C.4 – RSearch Solver Interface ......................................................................... 244

## LIST OF ACRONYMS

| | |
|---|---|
| CP | Compromising programming |
| DD | Dominated degree metric |
| DI | Diversity metric |
| DM | Decision maker |
| DO | Dissolved oxygen |
| DP | Dynamic programming |
| DR | Dominated ratio metric |
| EA | Evolutionary algorithm |
| GA | Genetic algorithm |
| GD | Generational distance metric |
| ICC | Interactive Compromise Coordinate |
| iGD | Inverted generational distance metric |
| KKT | Karush-Kuhn-Tucker |
| MCDA | Multi-criteria decision analysis |
| MCDM | Multiple criteria decision making |
| MO | Multi-objective optimization |
| MOEA | Multi-objective evolutionary algorithm |
| MOPSO | Multi-objective particle swarm optimization |
| NSGA | Non-dominated sorting genetic algorithm |
| PSO | Particle swarm optimization |
| PT | Processing time metric |
| SP | Spacing metric |
| T | Temperature |
| TDS | Total dissolved solids |
| VBA | Visual Basic for Applications |
| VEGA | Vector evaluated genetic algorithm |

# 1. INTRODUCTION

Section 1 of the United States Flood Control Act of 1936 declared flood control as a proper activity of the Federal Government and stipulated that Federal involvement would be justifiable "*if the benefits to whomsoever they may accrue are in excess of the estimated costs*". This is frequently referred to as the origin of benefit-cost analysis in public investments. For many years, decision making in water resources was essentially driven by the estimation of national benefits and costs.

Since the rise of the environmental movement in the 1960s new societal values and norms have emerged requiring the explicit consideration of multiple objectives in decisions regarding the use and management of natural resources in general, and particularly of water resources. Indeed, water resources problems are inherently multi-objective. Water not only serves different purposes like human and animal consumption, food production, energy production, recreation, maintenance of natural habitats, navigation, etc., but also invokes values of very distinct natures. Decision making in water resources may frequently require the consideration of economic, social, historical, cultural, technological, and other perspectives that are very difficult to commensurate in a single objective.

Economic valuation methods can be used to capture different values of water resources. Some values can be derived from traditional market-based approaches while others will require the use of more recently developed non-market valuation techniques. When the use of such methods is possible and doable, multiple objectives may be reduced to the common ground of economic returns and optimal solutions can still be searched by optimizing these returns. This is not always the case, however. These methods require additional information that may not be available or could not be practically obtainable. Moreover, some of these methods are not widely accepted yet, and many decision makers are still reluctant to rely on them.

Optimization techniques provide a way to determine the optimal trade-off regions so that multiple objectives can be analyzed. In single-objective optimization, an algorithm is used to find the global optimum by searching solutions in the space spanned by the decision variables, within a feasible region defined by the constraints imposed on the problem. In multi-objective optimization, in addition to the decision space, an objective space is introduced, spanned by the vector of objective functions. The concept of global optimum is now meaningless, if the objectives conflict. In multi-objective analysis, a set of non-dominated solutions is usually produced instead of a single optimum. According to the concept of non-dominance[1], also referred to Pareto optimality, a solution to a multi-objective problem is non-dominated, or Pareto optimal, if and only if no objective can be improved without worsening at least one other objective. These non-dominated solutions usually form a Pareto front in the objective space. Different Pareto fronts for a two-objective problem are presented in Figure 1.1 (Deb 2001).

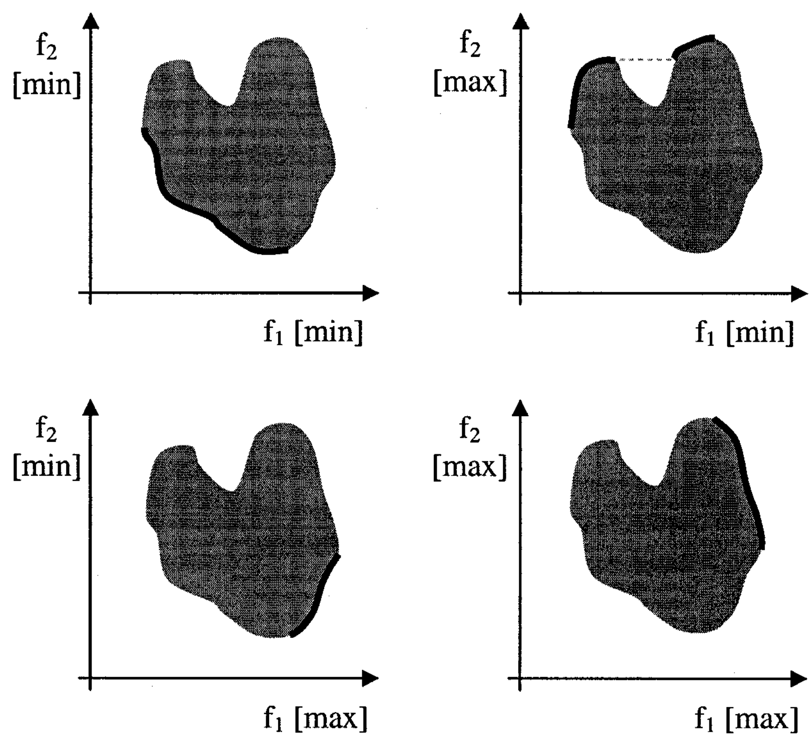



Figure 1.1 – Pareto Fronts

[1] A formal definition of the concept is provided later in this dissertation.

The multi-objective optimization normally involves two goals: (i) finding solutions as close as possible to the true Pareto front; and (ii) finding solutions that reasonably cover the whole extension of the Pareto front. The first goal is analogous to the goal of finding the global optimum in the single-objective case. The second goal assures that a wide range of trade-offs is represented in the non-dominated set of solutions, thus providing the decision maker (DM) with more information to help them select a specific solution.

Many methods have been formulated for multi-objective analysis. Cohon and Marks (1975) classified multi-objective methods in three categories: (i) generating techniques (with a posteriori articulation of preferences); (ii) techniques which rely on prior articulation of preferences; and (iii) techniques which rely on progressive articulation of preferences. This is still one of the most commonly used classifications (Goicoechea et al. 1982, Coello Coello et al. 2002). Methods in the first category solve the multi-objective problem by trying to find the whole set of non-dominated solutions. DM's preferences are introduced in a second step to identify best-compromise solutions. Methods with prior articulation of DM's preferences convert the multiple objective functions into a single function which can be solved using a single-objective optimizer. Methods with progressive articulation of preferences start by finding a non-dominated solution. The solution is presented to the DM and the DM's reactions are used to modify the optimization problem. These two steps are repeated until a satisfactory solution is obtained. The later approach requires intensive participation of the DM.

Deb (2001) refers to the generating methods as the ideal approach since they provide a more effective and comprehensive decision support information. The generating methods are also suitable when taking into consideration Herbert A. Simon's concept of "satisficing" behavior, where decision makers usually look for satisfactory alternatives rather than best, optimal solutions (Simon 1957). A set of solutions with their respective trade-offs would then convey more valuable information to these decision makers than a single recommended optimal solution.

Traditional multi-objective optimization methods attempt to generate the set of non-dominated solutions using mathematical programming. In the case of nonlinear problems, the weighting method and the ε-constraint method are the most commonly used techniques. Both methods transform the multi-objective problem into a single-objective one which can be repeatedly solved using nonlinear optimization to find the Pareto set.

With the weighting method, non-dominated solutions are obtained by parametrically varying the weights applied to the objective functions, however not all Pareto optimal solutions can be found unless all objective functions as well as the feasible region are convex. Another disadvantage of this method is that many different sets of weights may produce the same solution, compromising the efficiency of the method. When the weights reflect the preferences of the DM, the optimization gives the best-compromise solution, i.e. the solution which produces the highest utility to the DM. The ε-constraint method, on the other hand, does not require convexity but only leads to non-dominated solutions if certain specific conditions are satisfied (Miettinen 2001).

According to Coello Coello (2001), the first hints on the potential of evolutionary algorithms (EA) for multi-objective optimization occurred in the 1960s but the subject remained largely unexplored until mid-1980s. This author highlighted two advantages of evolutionary algorithms that make them particularly suitable for multi-objective optimization, when compared to traditional mathematical programming techniques:

- EA work simultaneously with a set of solutions, the so-called population, and several non-dominated solutions may be found in a single run of the algorithm;
- EA are less sensitive to the shape or continuity of the Pareto surface.

Particle swarm optimization – PSO (Kennedy and Eberhart 1995) is one of the newest techniques within the family of evolutionary optimization algorithms. The algorithm is based on an analogy with the choreography of a flight of a flock of birds. Due to its fast convergence, PSO has been advocated to be especially suitable for multi-objective optimization. Indeed, a number of multi-objective PSO (MOPSO) algorithms have recently been proposed in the literature; however, very few applications to real-world problems have been reported.

The objective of this research was two-fold: (i) to investigate the applicability of MOPSO for selected water resources multi-objective problems, evaluating advantages and limitations it might have by comparison to mathematical programming and other evolutionary algorithms; (ii) to investigate ways of displaying results of multi-objective analyses where the preferences of the decision makers can be interactively introduced.

Six chapters follow this introduction. A brief literature review covers basic concepts, mathematical programming, evolutionary multi-objective optimization in water resources, and multi-objective particle swarm optimization. The methodological aspects are discussed in the third, the implemented multi-objective solvers are described in the fourth, and the Interactive Compromise Coordinate method is introduced in the fifth chapter. The applications are presented in the sixth chapter, followed by the conclusions.

# 2. LITERATURE REVIEW

Since the 1960s a large number of techniques for solving multi-objective problems have been proposed. In the mid-1970s, Cohon and Marks (1975) could already identify more than 20 different methods in mathematical programming, and a large number of variations of these methods have been proposed as well. Mathematical programming methods are usually classified in more recent literature as the classical approaches.

With the rise of multi-objective evolutionary algorithms in the 1980s, new methods were developed. Several multi-objective extensions of the genetic algorithm (GA) have been introduced, starting with David Shaffer's vector evaluated GA in 1984 (Deb 2001). Other approaches have also been proposed since the late 1990s, like the multi-objective extensions of memetic algorithms, ant colony methods, and particle swarm optimization (Coello Coello et al. 2002).

It was not an objective of this research to perform an extensive review of all these methods. Three main areas are here reviewed in more detail: (i) mathematical programming, with emphasis to the most commonly used generating methods, and also some approaches with prior articulation of preferences; (ii) use of evolutionary multi-objective optimization in water resources; and (iii) multi-objective particle swarm optimization.

## 2.1. Notation and Basic Concepts

A general multi-objective optimization problem may be written as follows:

$$\left.\begin{array}{lll} \text{Maximize} & f_k(\vec{x}), & k = 1,2,\ldots,p \\ \text{subject to} & g_j(\vec{x}) \le 0, & j = 1,2,\ldots,m \end{array}\right\} \tag{2.1}$$

The optimization problem is to find the vector(s) $\vec{x}^* = [x_1^*, x_2^*, \ldots, x_n^*]$ which solves (2.1)[2]. The constraints $g$ in (2.1) define the feasible region $\Omega$. An objective function $f_i$ to be minimized can be converted to a maximization format by multiplying by -1. Constraints in a "$g \ge 0$" format can also be converted to the "$g \le 0$" format of (2.1) by multiplying by -1. Equality constraints "$g = 0$" can be defined by introducing two constraints, i.e. "$g \le 0$" and "$-g \le 0$".

[2] Normally several solutions will exist. Some authors use the notation *Max-dom* instead of Maximize to indicate that Pareto optimality is being searched.

Formal definition of Pareto dominance:

A solution vector $\vec{u} = [u_1, u_2, \ldots, u_p]$ is said to dominate a solution vector $\vec{v} = [v_1, v_2, \ldots, v_p]$ (also denoted by $\vec{u} \preceq \vec{v}$) if and only if

$$\forall i \in \{1,\ldots,p\},\ u_i \geq v_i \wedge \exists i \in \{1,\ldots,p\}: u_i > v_i\ .$$

A feasible solution $\vec{u}$ is non-dominated (or non-inferior) if and only if there is no other feasible solution $\vec{v}$ such that $\vec{v}$ dominates $\vec{u}$ ($\vec{v} \preceq \vec{u}$). The Pareto optimal set is the set of all such non-dominated solutions, which may be defined as:

$$P^* := \left\{\vec{x} \in \Omega \,|\, \neg\exists\ \vec{x}' \in \Omega : \vec{f}(\vec{x}') \preceq \vec{f}(\vec{x})\right\} \tag{2.2}$$

Non-dominated solutions are defined in the objective space. A solution $\vec{x}$ in the decision space is called efficient if and only if $\vec{x}$ is mapped by the set of objective functions to a non-dominated vector in the objective space.

## 2.2. Mathematical Programming

This section covers classical methods to generate the Pareto set and to select optimal compromise solutions by incorporating the DM's preferences.

### 2.2.1. Generating Methods

The weighting method and the ε-constraint method are the two most commonly used mathematical programming techniques to generate Pareto sets in multi-objective problems. According to Cohon and Marks (1975), both methods were introduced to public investment planning by Stephen A. Marglin in 1962 and 1967.

*The Weighting Method*

The weighting method is a direct result of the Karush-Kuhn-Tucker (KKT) non-inferiority conditions: if a solution $\vec{x}^*$ to the problem in (2.1) is non-inferior, then there exist $w_k \geq 0$, $k = 1,2,\ldots,p$ ($w_r$ strictly positive for some $r = 1,2,\ldots,p$), and $\lambda_i \geq 0$, $i = 1,2,\ldots,m$, such that

$$\vec{x}^* \in \Omega \tag{2.3}$$

$$\lambda_i g_i\left(\vec{x}^*\right) = 0, \qquad i = 1,2,\ldots,m \tag{2.4}$$

$$\sum_{k=1}^{p} w_k \nabla f_k\left(\vec{x}^*\right) - \sum_{i=1}^{m} \lambda_i \nabla g_i\left(\vec{x}^*\right) = 0 \tag{2.5}$$

Condition (2.3) guarantees feasibility, (2.4) are the complementary slackness conditions, and (2.5) is the stationarity condition. These are necessary conditions for the non-inferiority of solution $\vec{x}^*$. They are both necessary and sufficient if all objective functions are concave and $\Omega$ is a convex set.

Zadeh (1963) showed that, given the third condition in (2.5), non-dominated solutions could be obtained by solving the following single-objective optimization problem, for different sets of parametrically varied weights:

$$Max \sum_{k=1}^{p} w_k f_k(\vec{x}) \tag{2.6}$$

Subject to:

$$\vec{x} \in \Omega \tag{2.7}$$

Where: $w_k \geq 0$ for all $k$, and strictly positive for at least one objective.

The weighting method has the main disadvantage of not being able to find non-dominated solutions located on non-convex portions of the Pareto front.

Some applications of the weighting method in water resources are presented in the Table 2.1.

Table 2.1 – Applications of the weighting method in water resources

| Reference | Description |
|---|---|
| Major (1969) | Pioneer work using weights to calculate benefit-cost ratios in multiple objective investment programs. |
| Laabs and Schultz (1992) | Weighting method with stochastic dynamic programming (DP) to find optimal operating rules for a reservoir with six objectives. |
| Georgakakos (1993) | Weighting method with extended linear quadratic Gaussian control method for operation of a three-reservoir system with two objectives. |
| Cardwell et al. (1996) | Weighting method with linear programming (LP) used to find minimum instream flow requirements for single-reservoir system with three objectives. |
| Ko et al. (1997) | Combination of weighting and ε-constraint methods with DP for optimal operation of pumps and reservoirs in service-water-transmission systems with three objectives. |
| Olsen et al. (2000) | Weighting method with LP for optimal floodplain management strategies with two objectives. |

*The ε-Constraint Method*

In this method one objective is maximized and the others are introduced as constraints as follows:

$$Max\ f_r(\vec{x}) \tag{2.8}$$

Subject to:

$$\vec{x} \in \Omega \tag{2.9}$$

$$f_k(\vec{x}) \geq \varepsilon_k, \qquad all\ k \neq r \tag{2.10}$$

Cohon and Marks (1975) showed that the ε-constraint method is also derived directly from the KKT third condition. The set of non-dominated solutions can be obtained by parametrically varying the lower bounds $\varepsilon_k$. The method does not require convexity but only provides Pareto solutions if certain conditions are satisfied. A solution $\vec{x}^*$ to the ε-constraint optimization problem is Pareto optimal if and only if: $\vec{x}^*$ is unique or it solves the problem for all $r = 1,2,\ldots,p$, where $\varepsilon_k = f_k(\vec{x}^*)$ for $k = 1,2,\ldots,p$, $k \neq r$ (Miettinen 2001).

Table 2.2 presents some applications of the ε-constraint method in water resources.

Table 2.2 – Applications of the ε-constraint method in water resources

| Reference | Description |
|---|---|
| Cohon and Marks (1973) | ε-constraint method with LP to optimize water resources allocation in a multiple-reservoir, multiple-water-use river basin with two objectives. |
| Yeh and Becker (1982) | ε-constraint method combined with LP and DP for optimal operation of a nine-reservoir system with five objectives. |
| Louie et al. (1984) | ε-constraint method with LP for basin-wide water resources management planning with three objectives. |
| Moy et al. (1986) | ε-constraint method with mixed-integer LP for reliability-based single-reservoir optimal operation with four objectives. |
| Shafike et al. (1992) | ε-constraint method to find non-dominated groundwater contamination management strategies and compromise programming and ELECTRE for ranking with three objectives. |
| Ko et al. (1992) | Application of ε-constraint and weighting methods with DP for optimal operation of a single-reservoir system with three objectives and ε-constraint with successive LP for a 6-reservoir, four-objective problem. Goal programming and compromise programming also used to select best-compromise solutions. |
| Kwanyuen and Fontane (1998) | ε-constraint and nonlinear branch-and-bound method for groundwater development planning with three objectives. |
| McPhee and Yeh (2004) | ε-constraint method with LP to find best groundwater pumping and recharge policies with three objectives. |

2.2.2. Methods with Prior Articulation of Preferences

Four of these methods are reviewed here: goal programming, for being one of the first and mostly used methods, PROMETHEE II, ELECTRE III, and Compromise

Programming, as examples of methods that are able to incorporate the DM's preferences, in order to select and rank alternative non-dominated solutions of the Pareto set.

*Goal Programming*

This method, originally introduced by Abraham Charnes and William W. Cooper in 1961 (Goicoechea et al. 1982), is based on the definition of targets, or aspiration levels, representing compromised ideals for each objective. The original objectives are expressed as soft constraints by introducing deviation terms that will constitute additional decision variables. The general weighted goal programming formulation may be written as follows:

$$Min \sum_{k=1}^{p} \frac{\left(w_k^+ d_k^+ + w_k^- d_k^-\right)}{T_k} \tag{2.11}$$

Subject to:

$$\vec{x} \in \Omega \tag{2.12}$$

$$f_k(\vec{x}) + d_k^- - d_k^+ = T_k, \qquad k = 1,2,\ldots, p \tag{2.13}$$

$$d_k^+, d_k^- \geq 0, \qquad k = 1,2,\ldots, p \tag{2.14}$$

The summation in (2.11) is usually called the achievement function. In this formulation the deviations are scaled (divided by target levels). The variables $d_k^-$ and $d_k^+$ represent deviations below and above the target, respectively. There exist many variations of the goal programming approach. In preemptive (or lexicographic) goal programming, the goals are solved one-by-one from the most to the least important.

The main disadvantage of goal programming is that it does not always provide non-dominated solutions. Depending on how the targets are set by the DM, the solutions obtained may be dominated (Goicoechea et al. 1982). Some goal programming applications in water resources are presented in Table 2.3.

Table 2.3 – Applications of the goal programming method in water resources

| Reference | Description |
|---|---|
| Can and Houck (1984) | Preemptive linear goal programming for real-time operation of a system with four multi-purpose reservoirs. |
| Loganathan and Bhattacharya (1990) | Five different goal programming approaches are used for multi-purpose, multi-period optimal reservoir operation in a system with four reservoirs. |
| Fontane and Schneider (1994) | Goal programming for optimal operation of selective withdrawal structure in a single reservoir with four goals. |
| Alidi and Al-Faraj (1994) | Linear goal programming for optimal operation of a desalination system with five goals. |
| Mao and Mays (1994) | Nonlinear goal programming for optimal management of freshwater inflows to an estuary with six goals. |
| Eschenbach et al. (2001) | Five case studies with preemptive linear goal programming for operation of basin-wide water resources system with eight reservoirs. |

*PROMETHEE II*

PROMETHEE II (Brans and Vincke 1985) works with pair-wise comparisons based on preference relations $P(a,b)$ which are fuzzy sets that provide a measure of the degree of truth to the assertion that alternative $a$ is preferred over alternative $b$.

There are six types of preference relations used in PROMETHEE II. The Type V (linear with indifference) is shown in Figure 2.1, for a maximizing criterion. A preference relation is defined for each criterion. The parameter $q$ defines an indifference region; if the difference in the evaluations of alternatives $a$ and $b$, $f_k(a)$-$f_k(b)$, for criterion $k$ is less than equal to $q$ the preference relation $P(a,b)$ is equal to zero, i.e. the two alternatives are indifferent or $b$ is preferred over $a$ under that criterion. If the difference in evaluations is greater than or equal to $p$ the preference relation $P(a,b)$ is equal to one, i.e. $a$ is preferred over $b$.

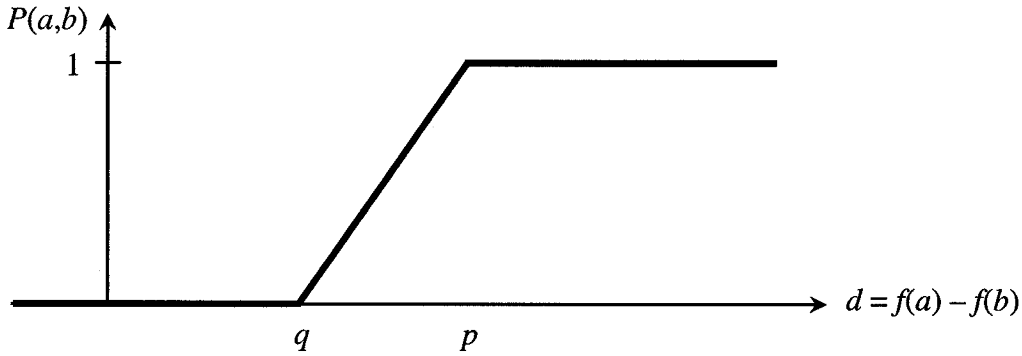



Figure 2.1 – PROMETHEE Preference Relation Type V

A fuzzy outranking is obtained through a global preference function $\pi(a,b)$ that is computed by aggregating the preference relations using weights (defined a priori) for each criterion. The global preference function is given by:

$$\pi(a,b)=\sum_{j=1}^{p} w_j P_j(a,b) \tag{2.15}$$

$$\sum_{j=1}^{p} w_j = 1 \tag{2.16}$$

Where: $p$ is the number of criteria.

With the global preference values the positive and negative outranking flows may be computed as follows.

$$\phi^{+}(a)=\sum_{x\in A}\pi(a,x) \tag{2.17}$$

$$\phi^{-}(a)=\sum_{x\in A}\pi(x,a) \tag{2.18}$$

Where: $A$ is the set of potential alternatives. The net outranking flow is given by:

$$\phi(a)=\phi^{+}(a)-\phi^{-}(a) \tag{2.19}$$

The net outranking flow provides a complete preorder where all alternatives are comparable to each other. Figure 2.2 presents a flowchart of the PROMETHE II method.

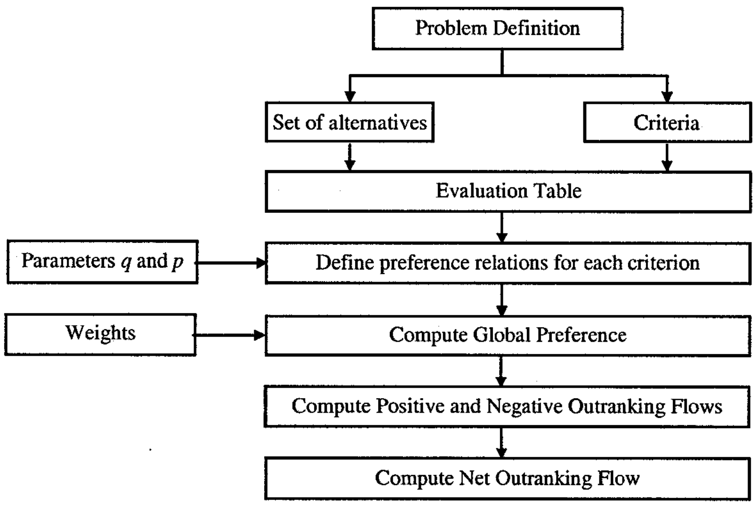



Figure 2.2 – PROMETHEE II Algorithm Flowchart

*ELECTRE III*

ELECTRE III, introduced by Bernard Roy in 1978 (Roy et al. 1992), also uses fuzzy relations to define preferences for each criterion. In ELECTRE III, however, the outranking relation $aSb$ measures the degree of truth to the assertion that "$a$ is at least as good as $b$".

ELECTRE III concordance relations are constructed for each criterion $k$ as follows.

$$c_k(a,b) = \begin{cases} 1 & \text{if} \quad f_k(b) - f_k(a) \leq q_k \\ 0 & \text{if} \quad f_k(b) - f_k(a) \geq p_k \\ \dfrac{p_k + f_k(a) - f_k(b)}{p_k - q_k} & \text{otherwise} \end{cases} \tag{2.20}$$

Global concordance is then computed over all criteria as follows.

$$c(a,b)=\frac{\sum_{k=1}^{p} w_k \cdot c_k(a,b)}{\sum_{k=1}^{p} w_k} \tag{2.21}$$

Where: $p$ is the number of criteria.

The discordance index $d_k(a,b)$ is computed as follows and then used to compute the credibility matrix $S(a,b)$ by adjusting (reducing) the global concordance values for those criteria where the discordance index exceeds the global concordance.

$$d_k(a,b)=\begin{cases} 0 & if \;\; f_k(b)-f_k(a)\le p_k \\ 1 & if \;\; f_k(b)-f_k(a)\ge v_k \\ \dfrac{f_k(b)-f_k(a)-p_k}{v_k-p_k} & otherwise \end{cases} \tag{2.22}$$

$$S(a,b)=\begin{cases} c(a,b) \;\; if\, d_k(a,b)\le c(a,b) \;\; \forall \; k \\ or: \;\; c(a,b)\cdot \prod_{k\in K(a,b)} \dfrac{1-d_k(a,b)}{1-c(a,b)} \end{cases} \tag{2.23}$$

Where: $K(a,b)$ is the set of criteria for which $d_k(a,b) > c(a,b)$, and $v_k$ is a veto threshold.

The way the discordance index is used when computing the credibility matrix gives ELECTRE III a non-compensatory characteristic which is a key difference between this method and PROMETHEE II.

Figure 2.3 presents a flowchart of ELECTRE III method.

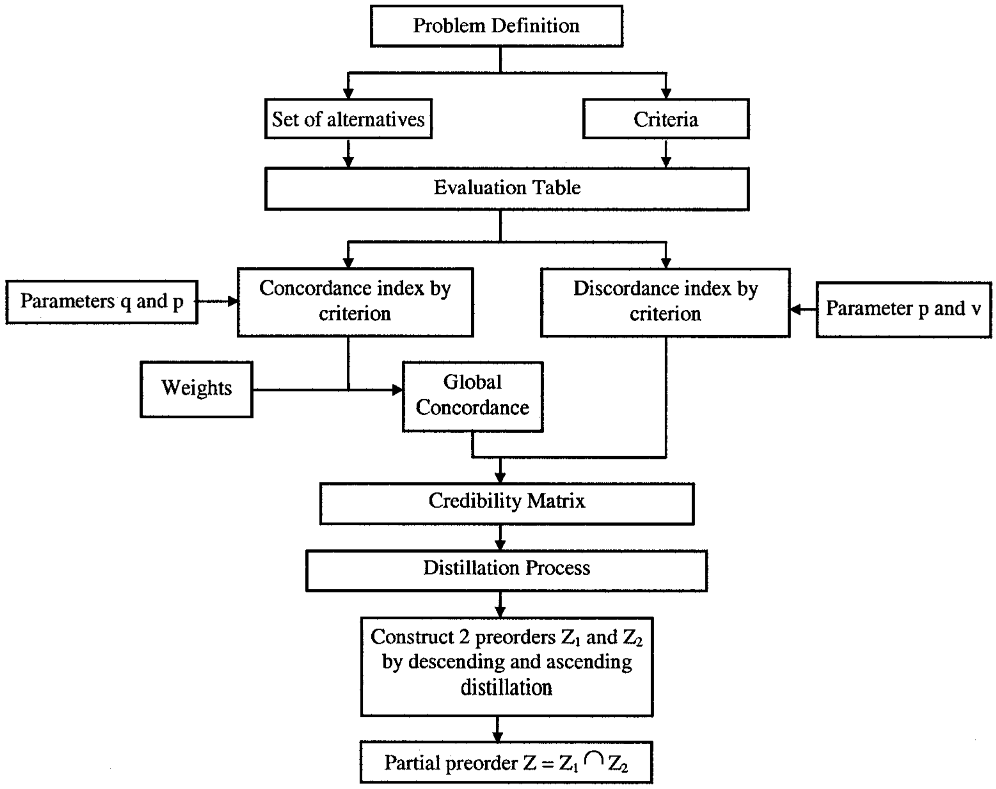



Figure 2.3 – ELECTRE III Algorithm Flowchart

*Compromise Programming*

Compromise Programming (CP), introduced by Milan Zeleny in 1973 (Goicoechea et al. 1982), follows a different approach. No fuzzy set is used to represent preferences. CP method uses distance measures from the ideal values for each criterion in order to obtain a ranking. Although different metrics can be used to measure these distances, the most commonly used are L-1 norm (absolute value criterion), L-2 norm (Euclidean norm), and L-$\infty$ norm (min-max or Chebyshev).

The following scaled norm is used to rank alternatives.

$$L_s(j) = \sum_{i=1}^{p} \alpha_i^s \cdot \left[ \frac{z_i^* - z_i(j)}{z_i^* - z_i^{**}} \right]^s \qquad (2.24)$$

Where: $s = 1$ (absolute value) or $s = 2$ (Euclidean norm); $p$ is the number of criteria; $z_i^*$ is the best value of $i$-th criterion over all alternatives; $z_i^{**}$ is the worst value of $i$-th criterion over all alternatives.

The Chebyshev scaled norm is given by:

$$L_\infty(j) = \underset{All\ i}{Max}\ \alpha_i \cdot \left[ \frac{z_i^* - z_i(j)}{z_i^* - z_i^{**}} \right] \qquad (2.25)$$

Each alternative $j$ will have a distance measured by its L-norm, and they will be ranked from the minimum norm to the maximum; i.e. the best alternative will be the one closest to the ideal. CP thus provides three different rankings, one for each norm. Further interaction with the decision maker will be needed if the rankings are too different. Figure 2.4 presents a flowchart of Compromise Programming method.

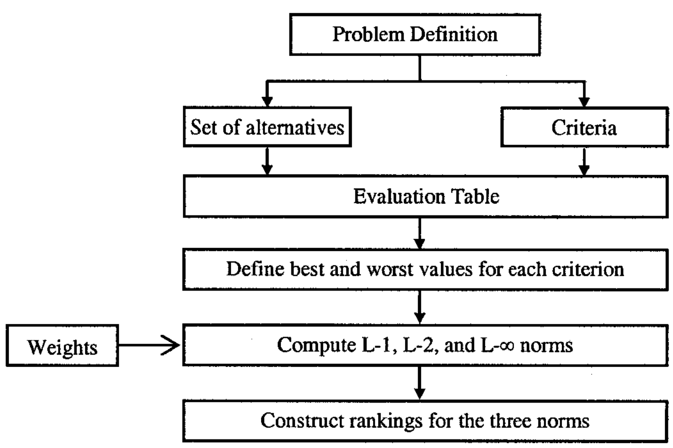



Figure 2.4 – Compromise Programming Algorithm Flowchart

Table 2.4 presents some applications of PROMETHEE, ELECTRE and Compromise Programming in water resources.

Table 2.4 – Applications of the PROMETHEE, ELECTRE and CP in water resources

| Reference | Description |
|---|---|
| Gershon and Duckstein (1983) | ELECTRE II and CP for ranking discrete water resources development alternatives in a river basin planning problem with thirteen criteria. |
| Tecle et al. (1988) | ELECTRE I and CP for selection of wastewater management schemes with twelve criteria. |
| Roy et al. (1992) | ELECTRE III for setting up priority order of water users based on seven criteria, for a rural water supply system. |
| Duckstein et al. (1994) | CP and ELECTRE III for ranking groundwater management alternatives with three criteria. |
| Cordeiro Netto et al. (1996) | ELECTRE III for selection of alternative actions for water resources development in a river basin with thirteen criteria. |
| Wen and Lee. (1998) | CP to find best-compromise solutions for water quality management in a river basin, using artificial neural networks to predict decision maker's preference structures. |
| Raju et al. (2000) | PROMETHEE II, ELECTRE III and CP for irrigation planning, ranking alternative strategies with ten criteria. |
| Hyde et al. (2004) | PROMETHEE II with reliability analysis based on Monte-Carlo simulation, presenting one case study on irrigation planning and another in water management both with ten criteria. |
| Elshorbagy (2006) | ELECTRE II and CP (L-1) to evaluate watershed modeling and monitoring alternatives seven criteria. |

## 2.3. Evolutionary Multi-Objective Optimization in Water Resources

Evolutionary algorithms (EAs) rely on analogies to natural evolutionary principles to build search and optimization procedures. The first developments in this area are associated with the genetic algorithms (GA) introduced by John H. Holland and his research group at the University of Michigan, Ann Arbor (Deb 2001).

These techniques consist of populations that are manipulated by a set of operators to move from one to the next generation. Each individual is an encoded solution which is evaluated by a fitness function.

EAs generally use three main operators: selection (or reproduction), recombination (or crossover in GA terminology), and mutation. Selection assures that the fittest individuals are the ones with highest probability to reproduce. Recombination defines what characteristics of the chosen individuals (parents) will be combined to form their next generation (children). Mutation changes some characteristics of an individual and has an important role in keeping diversity in the population, reducing the risk of a premature convergence to local optima. None of these operators are applied deterministically, which implies that poor individuals will eventually be generated. These individuals, however, will normally be eradicated in future generations by the selection operator. Figure 2.5 presents a flowchart of a general EA (modified from Deb 2001).

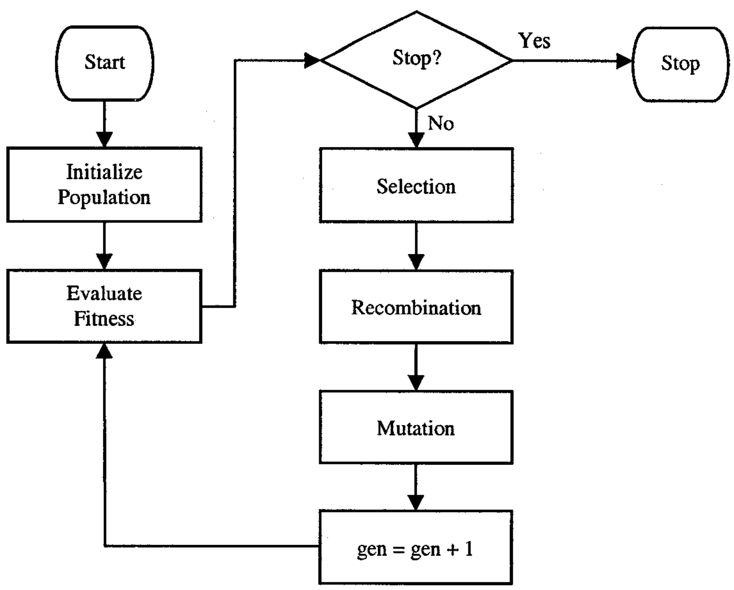



Figure 2.5 – Flowchart of a General EA

Although some suggestions towards the use of EA for multi-objective optimization appeared as early as the mid-1960s, the first real implementation of a multi-objective evolutionary algorithm (MOEA) was David Schaffer's vector evaluated genetic algorithm (VEGA) in 1984. VEGA worked with subpopulations associated with each objective. The algorithm was able to find non-dominated solutions but failed to keep a good distribution of solutions in the Pareto front (Deb 2001).

In 1989, David E. Goldberg suggested a new MOEA approach using Pareto dominance. In Goldberg's Pareto ranking approach, the original population is evaluated and non-dominated individuals receive the highest rank. The remaining population is again evaluated and the new non-dominated individuals are assigned the next highest rank. This is done sequentially until all individuals are ranked. The selection operator takes into account these rankings to favor reproduction of individuals with higher ranks. Goldberg also suggested a technique to promote diversity on the GA population. Based on Goldberg's ideas, a large number of different implementations of MOEAs have since been developed (Coello Coello 2002).

The non-dominated sorting genetic algorithm - NSGA (Srinivas and Deb 1995) follows Goldberg's suggestions using Pareto dominance to sort the population. It uses a different fitness assignment procedure and also a fitness sharing function method to keep diversity. The fitness sharing function penalizes the fitness of individuals located in crowded regions of the Pareto front, for each sorted class of non-dominated solutions. NSGA and its more efficient later version NSGA-II (Deb et al. 2002) have been used in a number of real-world applications, some of them reported in the water resources literature.

Many other MOEA methods exist which are not discussed in this dissertation. For a comprehensive review of such methods the reader is referred to Deb (2001) and Coello Coello (2002). Some applications of MOEA methods in water resources are presented in Table 2.5.

Table 2.5 – Applications of multi-objective evolutionary algorithms in water resources

| Reference | Description |
|---|---|
| Ritzel et al. (1994) | VEGA and Pareto ranking for optimal planning of a groundwater pollution hydraulic containment system, with two objectives. Results are compared to an ε-constraint approach. |
| Cieniawski et al. (1995) | VEGA, Pareto ranking, and a combination of the two methods for optimal design of groundwater monitoring system with two objectives. Results are compared to a weighting method approach. |
| Halhal et al. (1997) | A multi-objective GA for optimal water distribution network rehabilitation with two objectives. |
| Yeh and Labadie (1997) | VEGA and two different Pareto ranking approaches for optimal design of storm water detention system with two objectives. |
| Gupta et al. (1998) | Pareto ranking approach for multi-objective calibration of hydrologic models with three objectives. |
| Burn and Yulianti (2001) | Pareto ranking with fitness sharing approach in a waste-load allocation problem with two objectives. |
| Prasad and Park (2004) | NSGA-II with Pareto-based constraint handling technique for design of water distribution network with two objectives. |
| Prasad et al. (2004) | NSGA-II to define location and schedule to inject disinfectant in a water distribution network with two objectives. |
| Kapelan et al. (2005) | A multi-objective GA for optimal sampling design for calibration of a water distribution systems hydraulic model with two objectives. |
| Muleta and Nicklow (2005) | Strength Pareto Evolutionary Algorithm for selection of hydrologic units in a watershed management problem with three objectives. |
| Suen et al. (2005) | NSGA-II for optimal reservoir operation rules with two objectives. |
| Tang and Reed (2005) | Three MOEAs used for optimal calibration of semi-distributed hydrologic models with three objectives. |
| Farmani et al. (2005) | NSGA-II for a water distribution network rehabilitation problem with two and three objectives. |
| Atiquzzaman et al. (2006) | NSGA-II for optimal design of a water distribution network with two objectives. |
| Yandamuri et al. (2006) | NSGA-II with a gradually varied flow water quality model for optimal waste load allocation in rivers with two objectives. |

## 2.4. Multi-Objective Particle Swarm Optimization

Particle swarm optimization, introduced by James Kennedy and Russell C. Eberhart in 1995 (Kennedy and Eberhart 1995), is based on an analogy with the choreography of flight of a flock of birds. It is a population-based algorithm in which optimal solutions are searched through a combination of individual learning and social behavior.

There are many variants of the single objective PSO but in most of them the movement of the particles towards the optimum is governed by equations similar to the following:

$$\vec{v}_i(t+1) = w \cdot \vec{v}_i(t) + c_1 \cdot r_1 \cdot \left(\vec{P}_i(t) - \vec{x}_i(t)\right) + c_2 \cdot r_2 \cdot \left(\vec{P}_g(t) - \vec{x}_i(t)\right) \tag{2.26}$$

$$\vec{x}_i(t+1) = \vec{x}_i(t) + \vec{v}_i(t+1) \tag{2.27}$$

Where: $w$ is an inertia coefficient that has an important role balancing global (a large value of $w$) and local search (a small value of $w$), $c_1$ and $c_2$ are constants (usually $c_1 = c_2 = 2$), $r_1$ and $r_2$ are uniform random numbers in [0,1], $P_i$ is the best position vector of particle $i$ so far (personal best), $P_g$ is the best position vector of all particles so far (global best), $x_i(t)$ is the current position vector of particle $i$, and $v_i(t)$ is the current "velocity" of particle $i$. Mendes et al. (2004) suggest an inertia coefficient $w$ of less than 1, while other authors recommend to start with larger values and decrease with time, for example from a value of 1.4 to 0.5 (e.g. Elbeltagi et al. 2005, Jung and Karney 2006). Coello Coello et al. (2004) highlighted the sensitivity of the standard PSO algorithm to the value of $w$ and proposed the introduction of a mutation operator that assures an adequate global search while keeping a small value of $w$ (suggested 0.4) which favors a refined local search.

In multi-objective PSO, multiple non-dominated solutions are usually sought. The main difference in the multi-objective approach is how the $P_i$ and $P_g$ vectors are defined (fitness evaluation), and given that these vectors are not unique anymore, what values of $P_i$ and $P_g$ are selected to be used in (2.26).

A number of multi-objective extensions of the particle swarm optimization algorithm have been proposed since the late-1990s. Most of them use Pareto dominance pair-wise comparisons or rankings to drive the search for non-dominated solutions. Most approaches differ basically in two aspects: (i) the way they promote diversity; and (ii) the way they select the personal and global bests used to update the particles' positions.

Moore and Chapman (1999) proposed a multi-objective PSO based on Pareto dominance comparisons. Non-dominated solutions were stored in a continuously updated list. No mechanism to promote diversity was used. Parsopoulos and Vrahatis (2002) followed a different approach to evaluate the fitness of individual particles. Instead of Pareto dominance, a weighted aggregation technique is used. The authors also introduced a VEGA-based PSO approach. Ray and Liew (2002) used Pareto ranking incorporating some ideas of NSGA, and introduced a Pareto-based constraint handling technique.

Coello Coello and Lechuga (2002) and Coello Coello et al. (2004) used an external repository to store non-dominated solutions, an adaptive grid approach to select the global best, and a mutation operator to further promote diversity[3]. Fieldsend and Singh (2002) also used Pareto dominance but with a different global best selection procedure for each particle based on closeness to members of the non-dominated set. Hu et al. (2003) followed a lexicographic approach where objectives are optimized one-by-one, and uses an extended memory (like a repository) to store Pareto solutions. Mostaghim and Teich (2003) used an approach similar to compromise programming (called the Sigma method) to select different global best particles for each particle in the population.

[3] Further developments of this approach were reported in Pulido and Coello Coello (2004), incorporating clustering techniques, Villalobos-Arias et al. (2005) suggesting new mechanisms to maintain diversity, and Lechuga and Rowe (2005), with an explicit fitness sharing function.

Li (2003) combined elements of NSGA-II with PSO. Pareto sorting is used and the personal bests, $P_i$ in (2.26), are updated by comparing not only with the particle's current position but also with all other particles' personal bests. Fitness sharing and crowding are used to maintain diversity.

Benitez et al. (2005) proposed a method based on Pareto dominance that does not use any distance-based mechanism to select global best particles from the non-dominated repository. It is suggested that the approach could avoid scaling problems that may occur when distance metrics are used to keep diversity, like in fitness sharing, crowding distance, and adaptive grids.

Pulido (2005) further developed the algorithm of Coello Coello et al. (2004), introducing the use of clustering techniques to divide the swarm into sub-swarms to improve the spread of solutions. This algorithm also includes a routine to adapt the MOPSO parameters (inertia coefficient $w$, and parameters $c_1$ and $c_2$) based on information of previous cycles. The author uses a number of test functions, most of them with two objectives, comparing the results of the algorithm to those of other MOPSO implementations and MOEA methods.

Huang et al. (2006) followed an approach that allows particles to learn from the other particles personal experiences, called comprehensive learning PSO. It included a learning probability parameter which decides whether a dimension of a particle learns from its own personal best ($P_i$) or from other particle's personal best in the current generation. The algorithm also uses an external repository and crowding to promote diversity.

Gill et al. (2006) used Pareto ranking and a proximity-based procedure to select global best particles from the non-dominated front (rank one). The algorithm also updates the position of the particles already in the non-dominated front by making them follow the "best" individual in the front, which is arbitrarily selected as the median of the Pareto front. The method is used to calibrate hydrological models with two objectives.

The main characteristics of the reviewed multi-objective PSO approaches are synthesized in Table 2.6. Many comparisons with other MOEAs were performed using several performance metrics and MOPSO generally ranked very high. Three observations should be made, as one can verify from Table 2.6: (i) most multi-objective PSO have been applied only to test functions; very few applications to real-world problems are reported in the literature; (ii) very few tests were made with three or more objective functions; and (iii) many approaches do not include any constraint handling mechanism.

Table 2.6 – Characteristics of selected multi-objective PSO approaches

| Characteristic | [1] | [2] | [3] | [4] | [5] | [6] | [7] | [8] | [9] | [10] | [11] | [12] |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Fitness evaluation scheme | | | | | | | | | | | | |
| Pure Pareto dominance | ✓ | | | ✓ | ✓ | | ✓ | | ✓ | ✓ | ✓ | |
| Pareto ranking | | | ✓ | | | | | ✓ | | | | ✓ |
| Other | | ✓ | | | | ✓ | | | | | | |
| Repository | ✓ | | | ✓ | ✓ | ✓ | ✓ | | ✓ | ✓ | ✓ | |
| Diversity scheme | | | | | | | | | | | | |
| Sharing function | | | ✓ | ✓ | | | | ✓ | | | | |
| Adaptive grid | | | | ✓ | | | | | | ✓ | | |
| Mutation | | | | ✓ | ✓ | | ✓ | | | ✓ | | |
| Crowding operator | | | | | | | | ✓ | | | ✓ | |
| Other | | | | | | | | | ✓ | ✓ | | |
| Test functions | ✓ | | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Real-world problem | | | ✓ | | | | | | | | | ✓ |
| Use with 3 or more objectives | | | | | | | ✓ | | ✓ | ✓ | | ✓ |
| Constraints | | | ✓ | ✓ | | | | | ✓ | ✓ | | ✓ |
| Comparison to other methods | | ✓ | | ✓ | ✓ | ✓ | ✓ | ✓ | | ✓ | ✓ | ✓ |

[1] Moore and Chapman (1999); [2] Parsopoulos and Vrahatis (2002); [3] Ray and Liew (2002); [4] Coello Coello and Lechuga (2002) and Coello Coello et al. (2004); [5] Fieldsend and Singh (2002); [6] Hu et al. (2003); [7] Mostaghim and Teich (2003); [8] Li (2003); [9] Benitez et al. (2005); [10] Pulido (2005); [11] Huang et al. (2006); [12] Gill et al. (2006).

# 3. RESEARCH STEPS, TEST PROBLEMS AND COMPARISON METRICS

The first step of this research was to develop the MOPSO Solver. The Solver was then applied to selected standard test functions proposed in the EA literature and also to selected water resources problems. The next step was to implement multi-objective solvers using the real-coded NSGA-II method[4], the ε-Constraint with Nonlinear Programming (ε-NLP), and a pure Random Search approach (RSearch). The Interactive Compromise Coordinate (ICC) method was also developed and included as part of the

[4] All variables will be defined as real numbers. There is also a binary-coded NSGA-II method.

MOPSO, NSGA-II, ε-NLP, and RSearch Solvers, as well as a stand-alone add-in to Excel. The solvers were applied to a set of test functions for which results from the application of other implementations of MOPSO and NSGA-II were available in the literature. These test functions were used for validation of the implemented codes and for comparison purposes as well. Common comparison metrics, the same number of function evaluations, and equivalent parameters were applied in order to compare the results of the solvers with those reported in the literature used as reference. Two new comparison metrics were proposed in order to provide a more intuitive interpretation of the results.

The solvers were also applied with three different water resources models, for which various problem setups were explored and discussed. Finally, the results were synthesized and advantages and limitations of MOPSO were discussed. A flowchart view of the methodology is presented in Figure 3.1.

### 3.1. Test Functions

Five test problems were selected from two main references: (i) the paper by Coello Coello et al. (2004) introducing the MOPSO algorithm that served as basis for the development of the MOPSO Solver implemented in this research, and (ii) the paper by Deb et al. (2002) where the NSGA-II algorithm was introduced. The NSGA-II code, in C language, is a free code available in the Internet, and was used as reference for the development of the real-coded NSGA-II Solver implemented here. In both papers, the results obtained through the algorithm being proposed are compared with results of other evolutionary algorithms. NSGA-II is one of the methods used by Coello Coello et al. (2004) to evaluate their implementation of MOPSO.

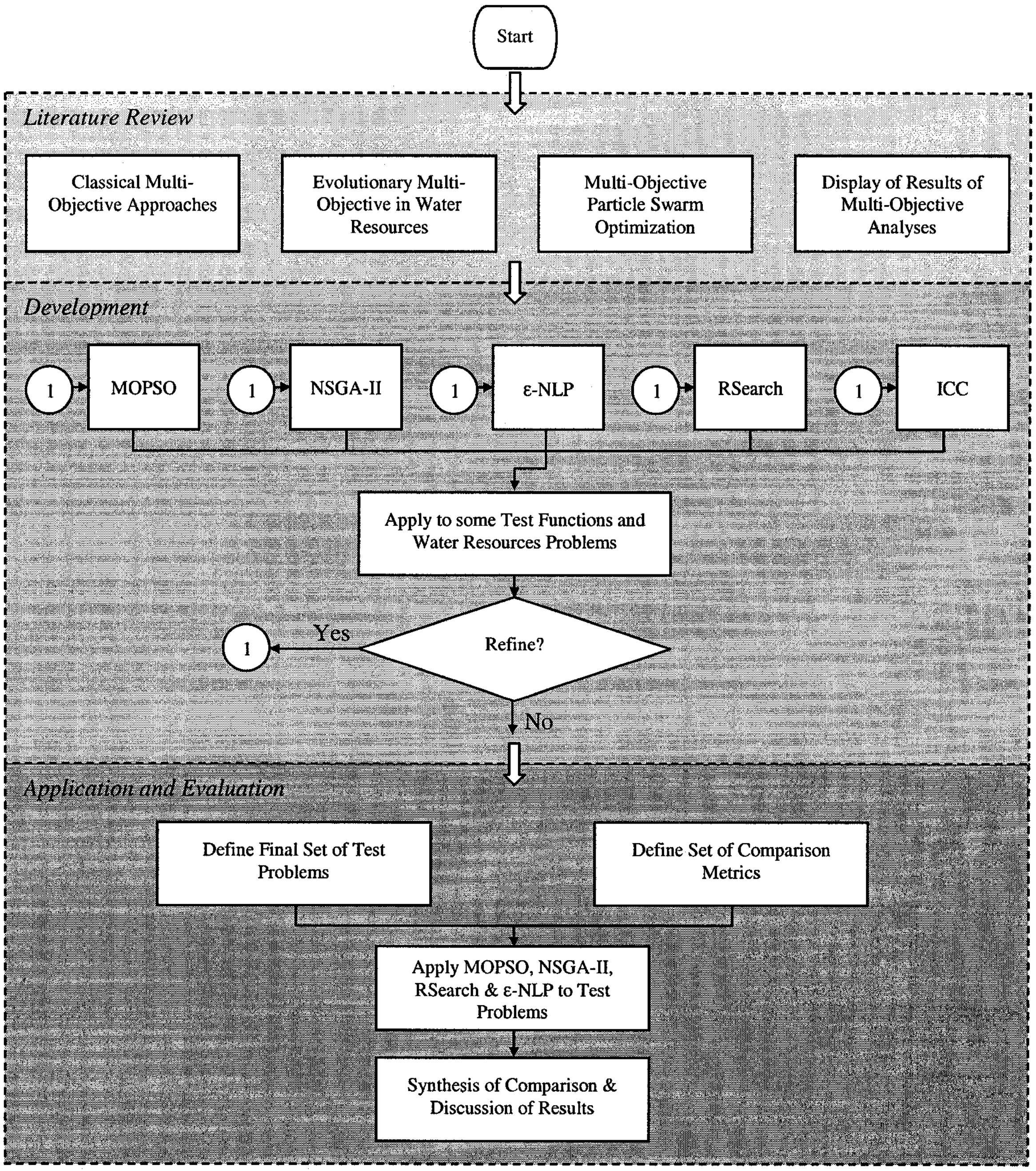



Figure 3.1 – Flowchart View of the Methodology

The set of five selected test problems is comprised of one unconstrained problem from each paper, one constrained problem from each paper, and the only problem discussed in both papers, which is also unconstrained. The problems present different characteristics and degrees of difficulty, which are described in Chapter 6.

### 3.2. Selected Water Resources Management Problems

Three different water resources management problems were selected to evaluate the performance of the MOPSO algorithm. The first one is a classic multi-purpose reservoir operation problem, the second is a problem of selective withdrawal from thermally stratified reservoirs, and the third is the operation of a multi-purpose reservoir using storage guide curves with fuzzy objectives. Different variants of these problems were solved involving two, three, four, and six objectives, and two, three, five, ten, and fourteen decision variables. The problems and the results obtained with the different methods are described in Chapter 6.

### 3.3. Comparison Metrics

In order to evaluate the performance of each algorithm, six performance metrics were applied. Four of them are usually used in the EA literature, and other two metrics were proposed.

The first three metrics were selected to allow comparisons with the results obtained by other authors with different algorithms. As previously mentioned, results obtained by Coello Coello et al. (2004) and Deb et al. (2002) were used for comparison and validation of the implemented solvers with the test functions mentioned in section 3.1. The metrics are: (i) generational distance (GD), employed in both references; (ii) spacing (SP), used by Coello Coello et al. (2004); and (iii) diversity metric (DI), used by Deb et al. (2002). An inverted generational distance (iGD) was also computed as suggested by Pulido (2005). Two other metrics were proposed, the dominated ratio (DR), which is similar to the error ratio introduced by Van Veldhuizen (1999), and the dominated degree (DD). In the next sections, the metrics are described and figures are included to demonstrate how they are computed. In all these figures a two-objective minimization problem is assumed.

### 3.3.1. Generational Distance (GD)

The generational distance was introduced by Van Veldhuizen and Lamont (1998) to measure the closeness of the resulting non-dominated solutions to the true Pareto front.

$$GD = \frac{\sqrt{\sum_{i=1}^{n} d_i^2}}{n} \tag{3.1}$$

where $n$ is the number of vectors in the test set, and $d_i$ is the Euclidean distance between each of those vectors and the nearest member in the known Pareto optimal front, as measured in the objective space. Figure 3.2 exemplifies the GD calculation. The smaller the GD the closer the test set is to the true Pareto optimal set.

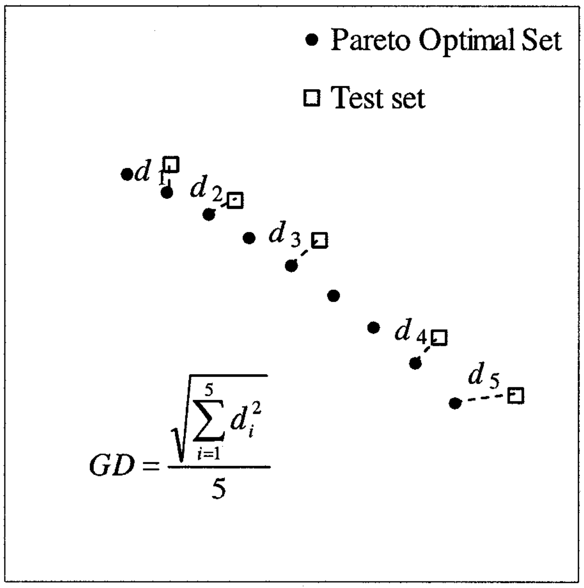



Figure 3.2 – Computation of the Generational Distance Metric

3.3.2. Spacing Metric (SP)

The spacing metric was proposed by Schott (1995) to measure the distribution of the potential non-dominated solutions.

$$S = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}\left(\bar{d} - d_i\right)^2} \tag{3.2}$$

where $d_i = \min_j \left( \left| f_1^i(\vec{x}) - f_1^j(\vec{x}) \right| + \left| f_2^i(\vec{x}) - f_2^j(\vec{x}) \right| \right)$, $i, j = 1, \ldots, n$, $\bar{d}$ is the mean of all $d_i$, and $n$ is the number of vectors in the test set. The SP metric is a surrogate measure of the distance variance of neighboring vectors in the test set. This metric does not include any information on the true Pareto optimal solutions. The calculation of the SP metric is demonstrated in Figure 3.3. The smaller the SP the more equally spaced the solutions in the test set will be in the objective space.

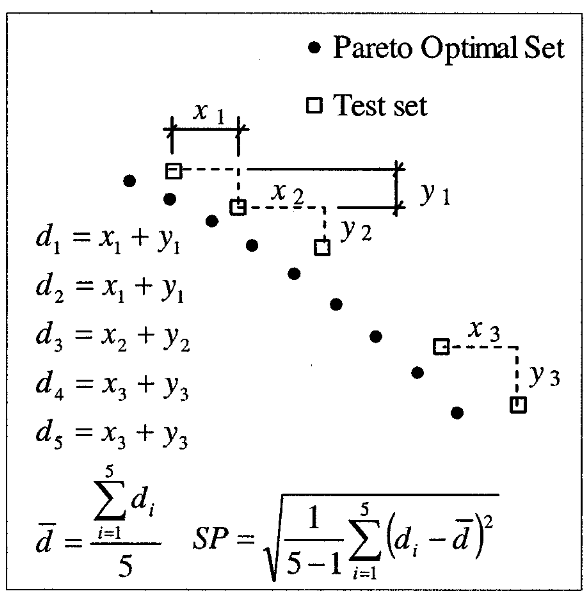



Figure 3.3 – Computation of the Spacing Metric

### 3.3.3. Diversity Metric (DI)

This metric was used by Deb et al. (2002) to measure the distribution and spread of the non-dominated solutions in the test set. The way it was defined the metric can only be applied in two-objective optimization problems.

$$DI = \frac{d_F + d_L + \sum_{i=1}^{n-1} \left| d_i - \bar{d} \right|}{d_F + d_L + (n-1) \cdot \bar{d}} \qquad (3.3)$$

Where $d_F$ and $d_L$ are the Euclidean distance between the extreme solutions on the test set and the known Pareto optimal set, $d_i$ are the Euclidean distances between two consecutive solutions in the test set, $i = 1, 2, \ldots, (n\text{-}1)$, $n$ is the number of solutions in the test set, a $\bar{d}$ is the average of all distances $d_i$.

If the extreme solutions of the test set are close to the extreme solutions on the known Pareto optimal set, and the intermediate solutions of the test set are evenly spaced the metric will have a value close to zero. For two test sets with similar $d_F$ and $d_L$, the one with the worst distribution of solutions will have a higher value of the DI metric. The calculation of the DI metric is demonstrated in Figure 3.4.

### 3.3.4. Inverted Generational Distance (iGD)

This metric was suggested by Pulido (2005) and it is similar to the generational distance but it measures the average Euclidean distance from the solutions on the known Pareto optimal set to the corresponding closest solution on the test set, instead. This was suggested as a way of capturing also some information on how well the test set covers the whole extension of the known Pareto optimal set.

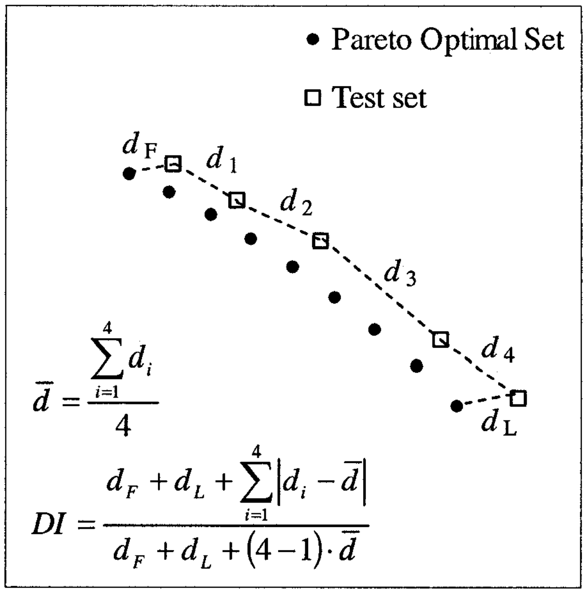



Figure 3.4 – Computation of the Diversity Metric

$$GD = \frac{\sqrt{\sum_{i=1}^{n} d_i^2}}{n} \qquad (3.4)$$

where $n$ is the number of vectors in the known Pareto optimal set, and $d_i$ is the Euclidean distance between each of those vectors and the nearest member in the test set. The computation of the iGD metric is exemplified in Figure 3.5.

Figure 3.6 shows three cases where excellent performance metrics can be misleading. In the first case to the left, the test set only covers a small fraction of the true Pareto optimal front, although the GD in this case would have an excellent value very close to zero. For this case the iGD would provide a more realistic measure of the performance. In the second case, the whole true Pareto optimal set is very well covered but the test set includes some bad solutions which are by far non-optimal. The iGD would still have an excellent value, however. The GD in this case would capture the real performance. In the last case, on the right side, solutions in the test set are far from the true Pareto optimal solutions and cover a much smaller range of the objective functions. The test set, however, would have an excellent value for the spacing metric. These cases serve to show the importance of analyzing the whole set of performance metrics, understanding the purpose and limitations of each one of them.

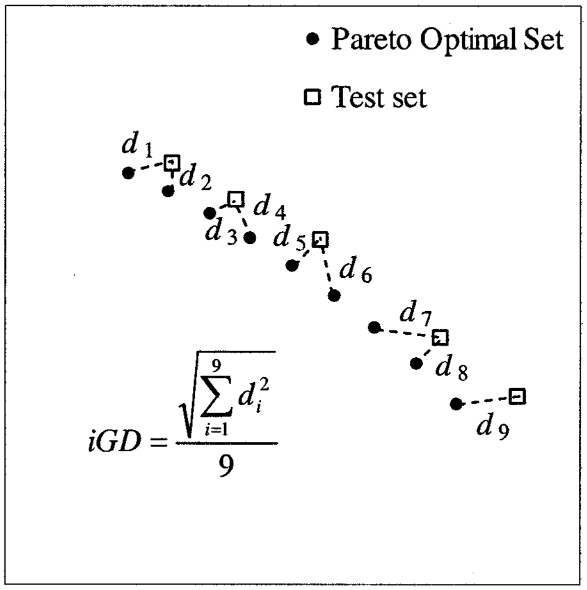



Figure 3.5 – Computation of the Inverted Generational Distance Metric

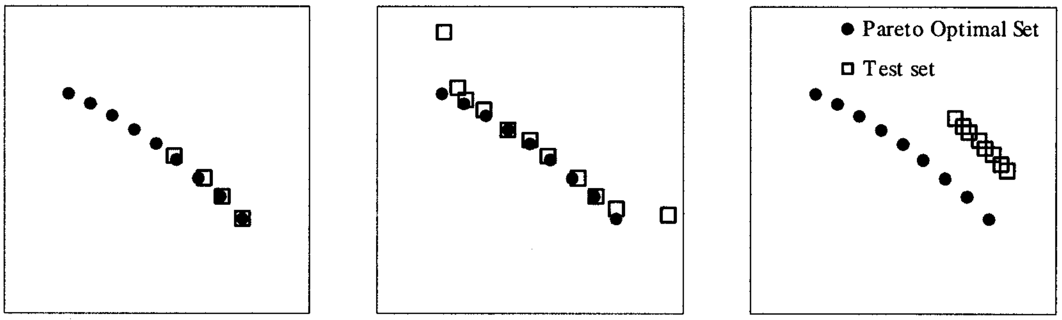



Figure 3.6 – Misleading Information from Performance Metrics

3.3.5. Dominated Ratio (DR)

This metric is proposed to measure the fraction of the solutions in the test set that are dominated by solutions in the true Pareto optimal set, or in the absence of the later, in the other set being compared. The smaller the DR the better the performance of the test set.

$$DR = \frac{n_d}{n} \tag{3.5}$$

where $n_d$ is the number of solutions in the test set that are dominated by any solution in the other set being compared, and $n$ is the total number of solutions in the test set. The DR computation is shown in Figure 3.7.

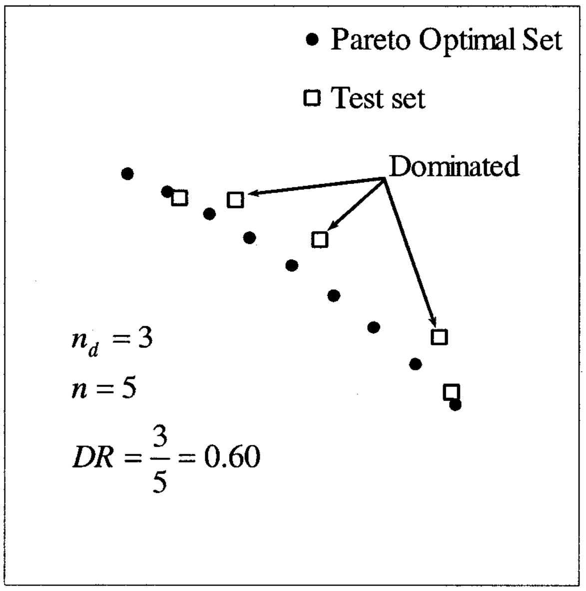



Figure 3.7 – Computation of the Dominated Ratio Metric

### 3.3.6. Dominated Degree (DD)

While the dominated ratio reflect the number of solutions in the test set that are dominated, the dominated degree measures how bad the dominated solution is relative to the maximum range of the objective values in the true Pareto optimal set. The dominated degree metric, in addition to the dominated ratio, was also used to make pair-wise comparisons between different sets, for example one obtained with MOPSO and the other with NSGA-II, when the true Pareto optimal solutions could not be obtained, in Test Problem TDT2 and FUZZY for example.

The dominated degree metric is given by:

$$DD = \frac{\sum_{j=1}^{n} dg_j}{n_d} \tag{3.6}$$

where $dg_j = \frac{\sum_{i=1}^{Nob} \frac{d_{j,i}}{R_i}}{Nob}$, $Nob$ is the number of objective functions, $d_{j,i}$ is the difference in the values of the objective $i$ between solution $j$ in the test set and the one in the true Pareto optimal set which dominates $j$, $R_i$ is the difference between the maximum and the minimum values of objective $i$ in the true Pareto optimal set, and $n_d$ is the number of solutions in the test set dominated by any solution in the Pareto optimal set (the product of the dominated ratio and the total number of solutions in the test set).

The computation of the DD metric, for the case with a known Pareto optimal set, is demonstrated in Figure 3.8. When the Pareto optimal set is unknown, two DD metrics are computed, the DD of Set 1 compared to Set 2 and the DD of Set 2 compared to Set 1; and the ranges $R_i$ of the objective values are taken from both sets combined.

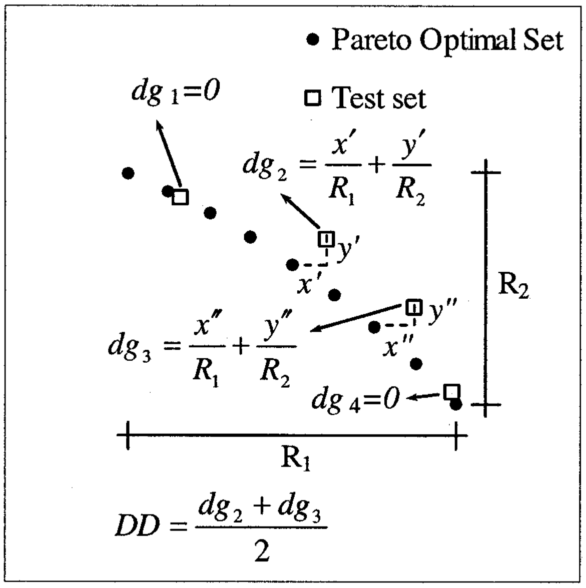



Figure 3.8 – Computation of the Dominated Degree Metric

# 4. THE METHODS: DESCRIPTION AND IMPLEMENTATION

Four multi-objective optimization methods were used in this research: (i) multi-objective particle swarm optimization (MOPSO); (ii) non-dominated sorting genetic algorithm II (NSGA-II), ε-constraint with nonlinear optimization (ε-NLP); and a pure random search method (RSearch). The interactive compromise coordinate (ICC) method, developed as part of this research for interactive display of alternative solutions of multi-objective problems, is described in the next chapter.

All methods were developed as add-ins to Microsoft Excel®. All the multi-objective solvers were developed with the same interface, where the user can setup the problem with much flexibility. The solvers accept up to six objectives. The interface for the MOPSO Solver is shown in Figure 4.1, while all the others are included in Appendix C.

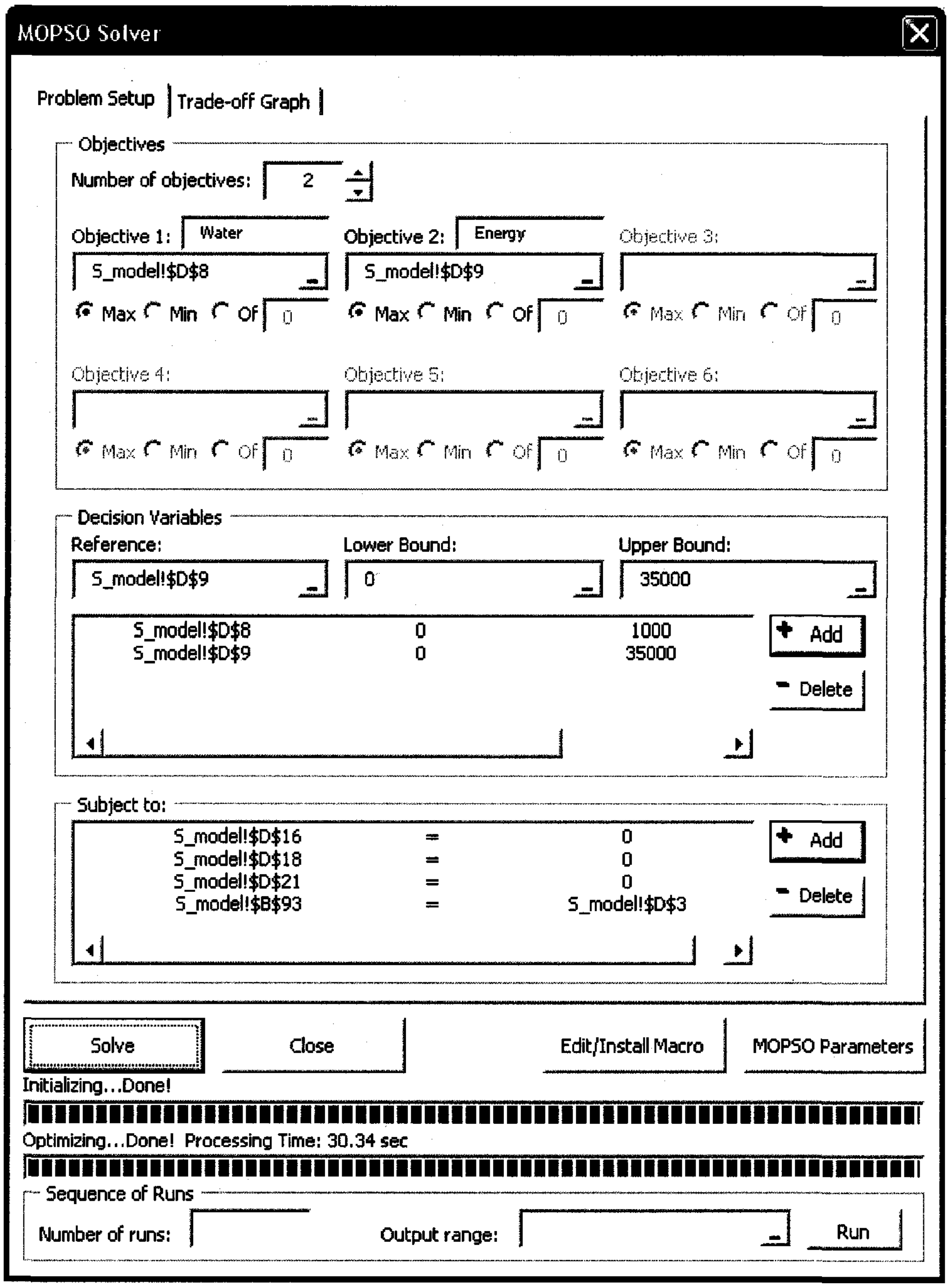



Figure 4.1 – MOPSO Solver Interface

The ICC method was imbedded in each solver to allow the analyst to visualize and explore the non-dominated solutions found by the solver (Figure 4.2). The ICC method also has a stand-alone interface, which can be called independently to visualize any set of solutions to a multi-objective optimization problem or multi-criteria decision analysis.

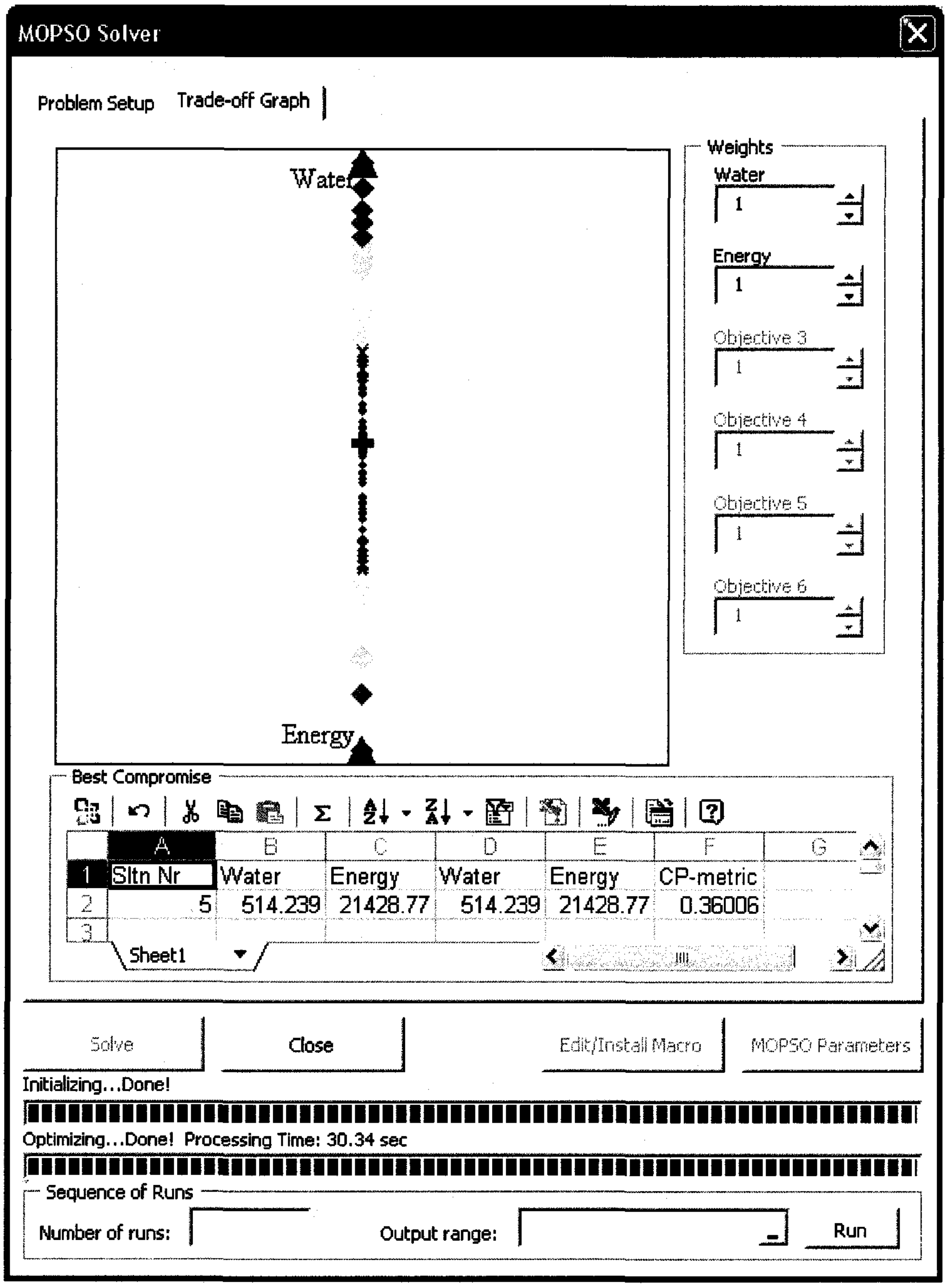



Figure 4.2 – ICC Method Imbedded in the MOPSO Solver

## 4.1. Multi-Objective Particle Swarm Optimization

A modified version of the algorithm proposed by Coello Coello et al. (2004) was used. The main characteristic of this algorithm is the use of an external repository which stores non-dominated solutions. The algorithm starts by generating an initial population. All the particles of this population are compared to each other and the non-dominated particles are stored in the repository. The particles' positions will be subsequently updated using the following:

$$\vec{v}_i(t+1) = w \cdot \vec{v}_i(t) + c_1 \cdot r_1 \cdot \left(\vec{P}_i(t) - \vec{x}_i(t)\right) + c_2 \cdot r_2 \cdot \left(\vec{R}_h(t) - \vec{x}_i(t)\right) \quad (4.1)$$

where: $R_h$ is a solution selected from the external repository in each iteration $t$, and the other terms are the same as defined in Eq. 2.26, with $w = 0.4$ and $c_1 = c_2 = 1.0$, as usual parameter values.

The best position vector of particle $i$, $P_i$, is initially set equal to the initial position of particle $i$. In the subsequent iterations, the best position vector is updated in the following way: (i) if the current $P_i(t)$ dominates the new position $x_i(t+1)$ then $P_i(t+1) = P_i(t)$, (ii) if the new position $x_i(t+1)$ dominates $P_i(t)$ then $P_i(t+1) = x_i(t+1)$, or (iii) if no one dominates the other then one of them is randomly selected to be the $P_i(t+1)$.

In MOPSO there is no such thing as the global best position vector ($P_g$) as in the standard PSO. There are several equally good non-dominated solutions stored in the external repository. To update the velocity of each particle using Eq. 4.1, the algorithm has to select one of the position vectors stored in the repository. As proposed by Pulido (2005), a combination of random and proximity-based selection is used, which was shown to perform better than either individual selection approaches. The first helps promote diversity while the second accelerates convergence. In all iterations, and for every particle in the population, the algorithm randomly defines the type of selection to be used with 50-50 chance for each. If the selection is based on proximity, the global best for a particle in the population will be the closest member of the current non-dominated set (in the objective space). The random selection is made in such a way that non-dominated solutions located in regions more densely populated in the objective space have lower probabilities of being selected, therefore leading to better distributions of points in the Pareto front. Instead of using the adaptive grid used in Coello Coello et al. (2004) and in Pulido (2005), the approach used in this research calculates, in the objective space, the density of points around each solution stored in the repository and performs a roulette wheel selection such that the probability of choosing one point is inversely related to its associated density. This approach is closer to the fitness sharing approach proposed by Lechuga and Rowe (2005).

In every iteration $t$, the new positions of all particles are compared among themselves and the non-dominated ones are then compared with all solutions stored in the repository. The repository is then updated, adding new non-dominated solutions and eliminating old solutions that are now dominated. The size of the repository is an important parameter to be set. Once the repository is full and a new non-dominated solution is found, then this new solution takes the place of another non-dominated solution in the repository which is selected randomly using a similar procedure based on density as described above but now assigning higher probabilities of being selected to solutions located in denser regions of the objective space. A mutation operator is also employed to enhance the global search capability of the MOPSO. The operator randomly changes the position in the decision space of a fraction of the particles in the population. This fraction is exponentially reduced with the number of cycles at a rate defined by the user. The algorithm runs until the maximum number of iterations (cycles) is reached.

The algorithm handles constraints in a very simple way. When comparing two different solutions, where at least one is infeasible, the algorithm does the following: (i) a feasible solution dominates an infeasible solution; (ii) with two infeasible solutions the one with smaller violation of the constraints dominates the other. To implement this procedure when several constraints are imposed, a scaled index is calculated to reflect the aggregated degree of constraint violation. The flowchart of the MOPSO Solver algorithm is presented in Figure 4.3.

### 4.1.1. MOPSO Solver Specific Features

Some specific features were developed to increase the flexibility of the MOPSO Solver and to allow the optimization of a particular type of problem commonly found in real-world applications.

#### *Scripting Tool for User-Defined Macros*

The MOPSO Solver includes a scripting tool which allows the user to code specific procedures that will be called immediately before the objective functions are evaluated in the MOPSO algorithm. This tool can be used to call external programs or perform computations that cannot be easily made in the spreadsheet. This considerably increases the flexibility of the MOPSO Solver. Figure 4.4 shows the MOPSO Solver screen where the user-defined macros can be written in VBA. This tool is also included in the NSGA-II and RSearch Solvers.

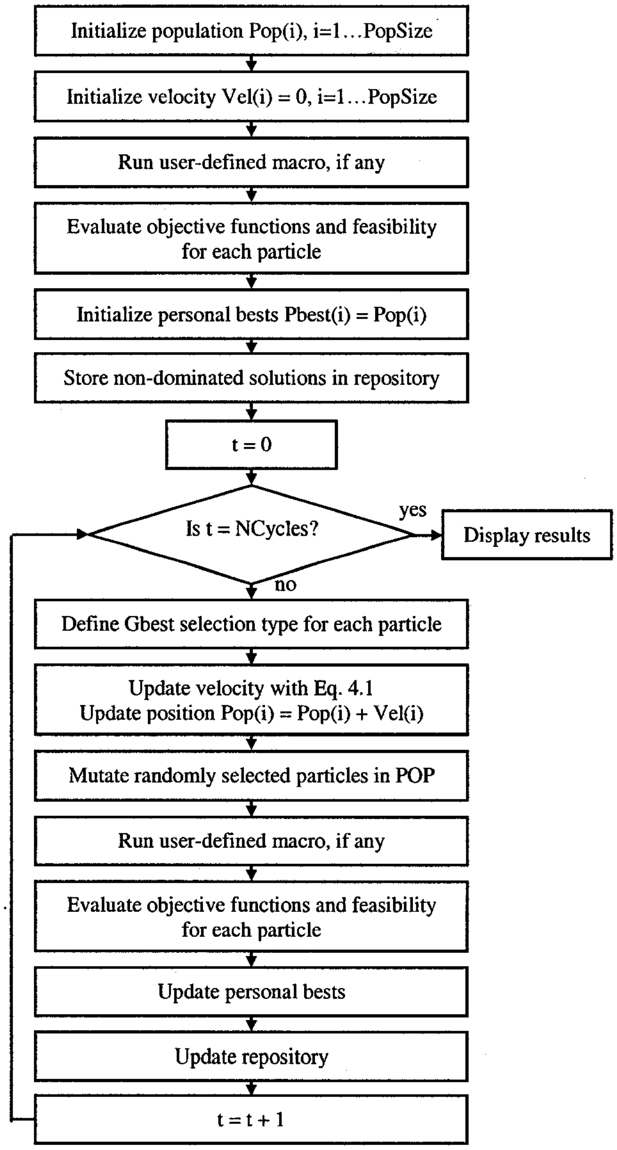



Figure 4.3 – Flowchart of the MOPSO Solver Algorithm

*Routine for Handling Equality Constraints Imposed on Decision Variables*

Multi-objective evolutionary algorithms cannot deal directly with equality constraints imposed to the decision variables, because of the random component of these methods. This limitation is discussed in section 6.2.2, which deals with the selective withdrawal problem. Populations are usually initialized by randomly generating values of all decision variables within their respective bounds. EA operators (like crossover, mutation, or the

MOPSO velocity update) are applied to move from one generation (or cycle) to the next. These operators are also randomly defined and the equality constraint would make the problem virtually unsolvable.

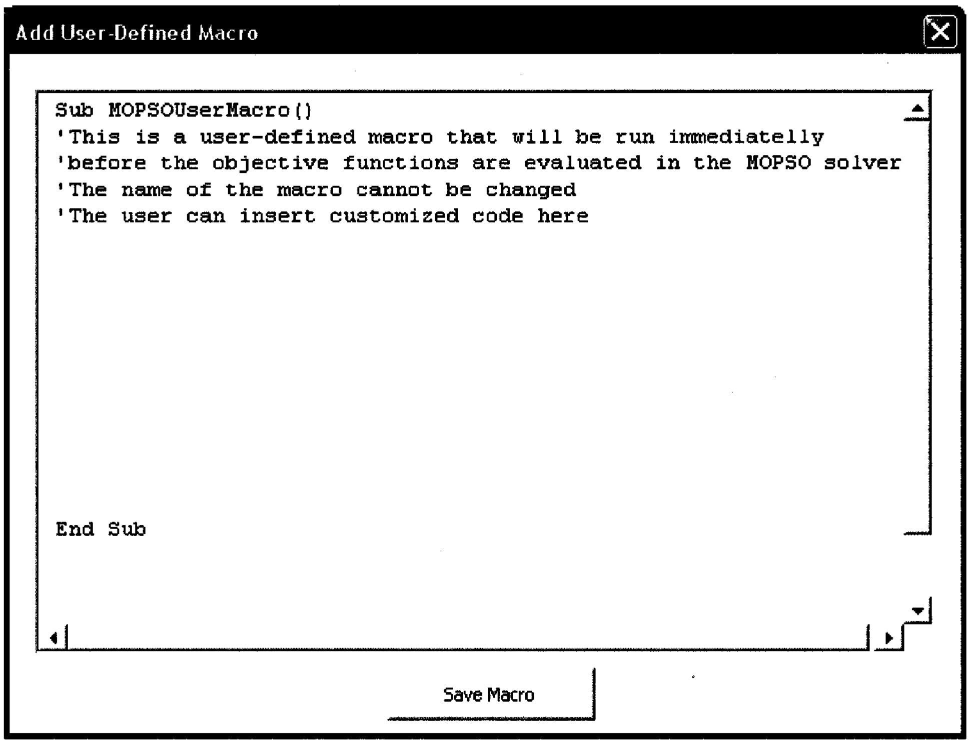



Figure 4.4 – MOPSO Solver Scripting Tool

A routine was included in the MOPSO Solver (and in the NSGA-II and RSearch Solvers as well) to deal with such equality constraints. This type of constraint was assigned a "sum=" operator in the "Add Constraint" dialog of the solvers (Figure 4.5).

Essentially, the routine allows the solver algorithm to define the decision variables in a sequential way, taking into account the decision variables previously defined, until the last decision variable, which will be deterministically defined as a function of the equality constraint. To implement this, a scheme of dynamic temporary bounds is used.

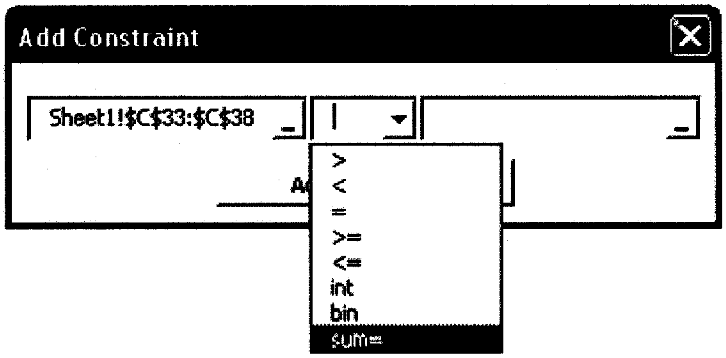



Figure 4.5 – MOPSO Solver Add Constraint Dialog

When no equality constraint is imposed, the decision variables are independently defined and the sequence in which that is done would not matter. This is not true when equality constraints are present. In this case, if the same sequence is always used, an artificial trend would be added to the definition of the variables. The following example shows why this can happen.

Let $X_1, X_2, \ldots, X_n$ be random variables uniformly distributed within $[0,u]$. The cumulative probability distribution function of the sum of such variables is given by the following equation (Uspensky 1937):

$$F(t,n) = \frac{1}{n!}\sum_{k \le \frac{t}{u}} (-1)^k \binom{n}{k}\left(\frac{t}{u} - k\right)^n \tag{4.2}$$

If $x_1+x_2+\ldots+x_n = u$ and the variables are defined sequentially from 1 to $n$, the following probabilities can be calculated:

$P(x_1=0) = 0$

$P(x_2=0) = P(x_1>u) = 0$

$P(x_3=0) = P(x_1+x_2>u) = 1 - F(u,n=2) = 1 - \frac{1}{2!}\left[(-1)^0 \cdot 1 \cdot (1)^2 + (-1)^1 \cdot 2 \cdot (0)^2\right] = 0.5$

The second term in the summation ($k$ = 1) will always be zero since $t = u$, then:

$$P(x_4{=}0) = P(x_1+x_2+x_3>u) = 1 - F(u,n{=}3) = 1 - \frac{1}{3!}\left[(-1)^0 \cdot 1 \cdot (1)^3\right] = \frac{5}{6} = 0.833$$

$$P(x_5{=}0) = P(x_1+x_2+x_3+x_4>u) = 1 - F(u,n{=}4) = 1 - \frac{1}{4!}\left[(-1)^0 \cdot 1 \cdot (1)^4\right] = \frac{23}{24} = 0.958$$

From this example, even with only five variables, one can see that the later variables in the sequence have high probabilities of being zero. MOEA methods usually use uniform random numbers to initialize populations and to update the values of the decision variables. If all individuals in the population (or particles in the swarm, in MOPSO terminology) are generated using the same sequence of decision variables, the last ones would be consistently zero.

To avoid this problem, the MOPSO, NSGA-II, and RSearch Solvers randomly select a sequence of decision variables for every individual or particle to be generated or updated by the algorithms. This approach is novel to the author's best knowledge. The pseudo-code of the algorithm for handling equality constraints imposed to decision variables is presented in Figure 4.6.

```
Ndim = number of decision variables
PopSize = population size
For j = 1 to PopSize
    Randomly define sequence Seq(q), where q = 1 to Ndim
    LocalUpperBound(q) = OriginalUpperBound(q), for q = 1 to Ndim
    LocalLowerBound(q) = OriginalLowerBound(q), for q = 1 to Ndim
    For q = 1 to Ndim
        Define value of decision variable Seq(q) based on LocalBounds(q)
        Update LocalBounds(q) {if next decision variable is the last to be defined,
            calculate its value based on the equality constraint and the previous decision
            variables, and make LocalUpperBound(q+1) = LocalLowerBound(q+1)}
    Next q
Next j
```

Figure 4.6 – Routine for Handling Equality Constraints Imposed on Decision Variables

*MOPSO Specific Operator for Extreme Solutions*

The tests performed with the basic MOPSO algorithm revealed that the method is generally very effective in finding non-dominated solutions in the whole extension of the Pareto optimal front. A problem was found to be recurrent, however. Many times the MOPSO algorithm would find a solution that is very close to the optimal for one objective but highly sub-optimal for other objective(s). This happens on the extremes of the Pareto fronts. This issue is exemplified in Figure 4.7, for a maximization problem.

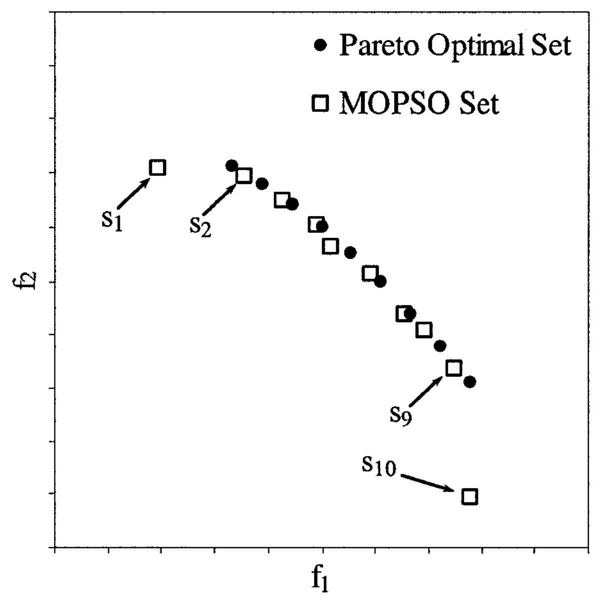



Figure 4.7 – Sub-Optimal Solutions on the Extreme of Pareto Fronts

MOPSO solution $s_1$ in Figure 4.7 is very close to the maximum possible value of $f_2$ but sub-optimal in terms of $f_1$, and the opposite applies to solution $s_{10}$. Since any solution in the repository can serve as global driver for particles in population, the algorithm will naturally tend to concentrate the particles in intermediate regions. The mutation operator improves the search capabilities in the extremes but excessive mutation would compromise convergence over the whole extension of the Pareto front.

Two approaches were implemented for dealing with this problem. The first consisted of performing a localized crossover in the extreme solutions for all objectives, every time the repository is updated. The crossover would be applied to solutions $s_1$ and $s_2$ and to solutions $s_9$ and $s_{10}$ in Figure 4.7. This approach worked well, especially with two-objective problems, but it is computationally expensive, with a significant increase in processing time. In addition, the results obtained for problems with three objectives (e.g. Test Problem WEF) were not as good.

The second approach included a post-processing routine, which tests all non-dominated solutions in the repository for possible Pareto improvements, at the end of all cycles. Each decision variable is varied in increasing and decreasing directions, while keeping the others unchanged. One of the following situations will occur:

(i) If a Pareto improvement is possible in only one direction, update the original position with the improved one;

(ii) If Pareto improvements are possible in both directions (i.e. both improved positions dominate the original one) then test if one improved position dominates the other, if so update the original with the non-dominated improved position, otherwise choose randomly one non-dominated improved position to update the original; and

(iii) If no Pareto improvements can be found, keep the original solution.

The pseudo-code for the MOPSO post-processing routine is presented in Figure 4.8. The approach worked well in most tests and it is less computational expensive than the crossover approach.

The post-processing routine cannot be used when equality constraints on decision variables ("sum=" type) are present, however. The post-processing routine changes one decision variable at a time, while keeping the others at original values and this cannot be done when equality constraints are imposed. An adaptation of both routines could be implemented to overcome this limitation, but this is left for future research.

```
Ndim = number of decision variables
RepSize = number of solutions in the repository
Range(q) = UpperBound(q) – LowerBound(q), {range of decision variable q}
Search directions: Sdir(1) = 1 (increasing), and Sdir(2) = -1 (decreasing)
For j = 1 to RepSize
  Randomly define sequence Seq(q), where q = 1 to Ndim
  For q = 1 to Ndim
    Define initial search step: Step(Seq(q)) = 0.001·Range(Seq(q))
    For k = 1 to 2 {k defines the search direction}
      CurrentPosition = OriginalPosition {for all dimensions in the search space}
      CanImprove(k) = False
      L = 1 {the step decreases from (0.01 to 0.001)·Range when L goes from 1 to 10}
      Do While L ≤ 10
        NewPosition(k,Seq(q)) = CurrentPosition(Seq(q))+Sdir(k)·Step(Seq(q))·(10/L)
        If  NewPosition is infeasible then
          L = L+1
        ElseIf  F(NewPosition) dominates F(CurrentPosition) then
          CanImprove(k) = True
          CurrentPosition(Seq(q)) = NewPosition(k,Seq(q))
        Else
          L = L+1
        EndIf
      Loop
    Next k
    If CanImprove(1) = True and CanImprove(2) = True then
        If  F(NewPosition(k=1)) dominates F(NewPosition(k=2) then
          OriginalPosition = NewPosition(k=1)
        ElseIf  F(NewPosition(k=2)) dominates F(NewPosition(k=1) then
          OriginalPosition = NewPosition(k=2)
        Else
          OriginalPosition = choose randomly NewPosition (k=1) or NewPosition(k=2)
        EndIf
    ElseIf  CanImprove(1) = True then  {can only increase}
      OriginalPosition = NewPosition(k=1)
    ElseIf CanImprove(2) = True then  {can only decrease}
      OriginalPosition = NewPosition(k=2)
    EndIf
  Next q
Next j
```

Figure 4.8 – Pseudo-Code of MOPSO Post-Processing Routine

4.1.2. MOPSO Parameters

The following parameters are adjustable in the MOPSO Solver: (i) population size, (ii) number of cycles, (iii) the repository size, (iv) inertia coefficient $w$, (v) the constants $c_1$ and $c_2$, that balance local and global search, (vi) the number of subdivisions used to implement the fitness sharing function, and (vii) the mutation rate, which determines the number of particles to be mutated in each cycle using Eq. 4.3.

$$F_m = \left(1 - \frac{t}{N_{cyc} \cdot MutRate}\right)^{1.5} \tag{4.3}$$

where $F_m$ is the fraction of the population to be mutated, $t$ is the current cycle (iteration), $N_{cyc}$ is the total number of cycles, and $MutRate$ is the mutation rate. Eq. 4.3 is applied if $t < N_{cyc} \cdot MutRate$, with $F_m = 0$ otherwise.

The default parameters of the MOPSO Solver are presented in Figure 4.9.

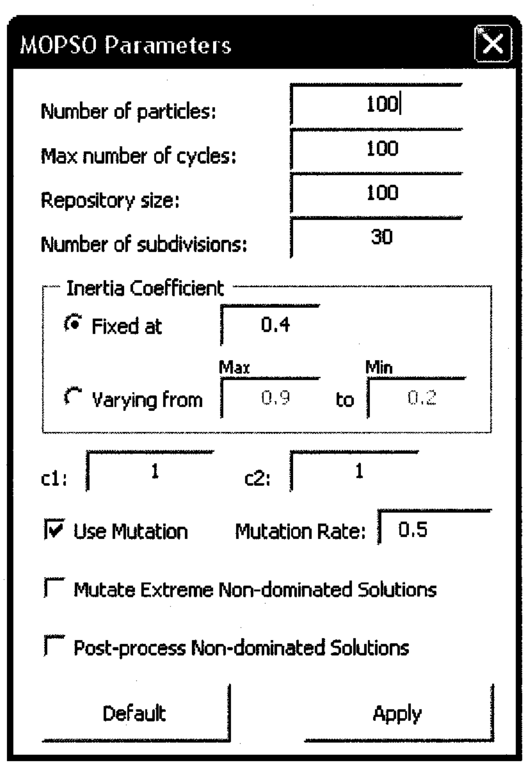



Figure 4.9 – MOPSO Solver Default Parameters

### 4.2. Non-Dominated Sorting Genetic Algorithm II

The NSGA-II was proposed by Deb et al. (2002) incorporating a number of improvements to the previous NSGA by Srinivas and Deb (1995). The improvements were basically three: (i) introduction of a faster sorting algorithm; (ii) use of elitism to improve convergence; (iii) change in the diversity-preserving mechanism from fitness sharing to a parameter-less crowded-comparison approach.

The new sorting algorithm starts by computing two entities: (i) the number of solutions that dominate solution $p$, called the domination count $n_p$, and (ii) the set $S_p$ formed by all solutions dominated by solution $p$. All solutions in the first non-dominated front will have $n_p = 0$. For each solution $p$ in the first front, the algorithm examines each associated dominated solution $q$ stored in set $S_p$ and reduces its domination count by one. If the domination count of $q$ becomes zero, $q$ is stored in a separate set $Q$. These solutions belong to the second non-dominated front. The procedure is continued with each member of set $Q$ so the third front is found, and the process continues to find all fronts.

Elitism is implemented by merging the parent and the offspring populations to form a new population twice as large. This population is sorted according to non-dominance and to crowding distance, and the best individuals are selected to be the parent population of the next generation.

The crowding distance assignment is performed in the objective space by sorting the population for each objective sequentially. Solutions with extreme values for any objective are assigned a crowding distance of infinity, so they will always be selected first. Intermediate solutions are assigned crowding distance values equal to the absolute normalized difference in the objective function values of two adjacent solutions.

Solutions with larger crowding distances are less crowded by other solutions and will be preferred when applying the selection operator. The pseudo-code for this procedure is shown in Figure 4.10.

```
Nob = number of objectives
Ndom = number of non-dominated solutions
S = the set of non-dominated solutions
For j = 1 to Ndom
  Sdist(j) = 0 {initialize crowding distance}
Next j
For i = 1 to Nob
  S = Sort (S, i) {sort all solution in S according to the value of objective i}
  Sdist(1) = Sdist(Ndom) = ∞
  For k = 2 to (Ndom – 1)
    Sdist(k) = Sdist(k) + [S(k+1, i) – S(k-1, i)]/(f_i^max – f_i^min)
  Next k
Next i
```

Figure 4.10 – Pseudo-Code of NSGA-II Crowding Distance Assignment Procedure

The NSGA-II procedure is synthesized in Figure 4.11.

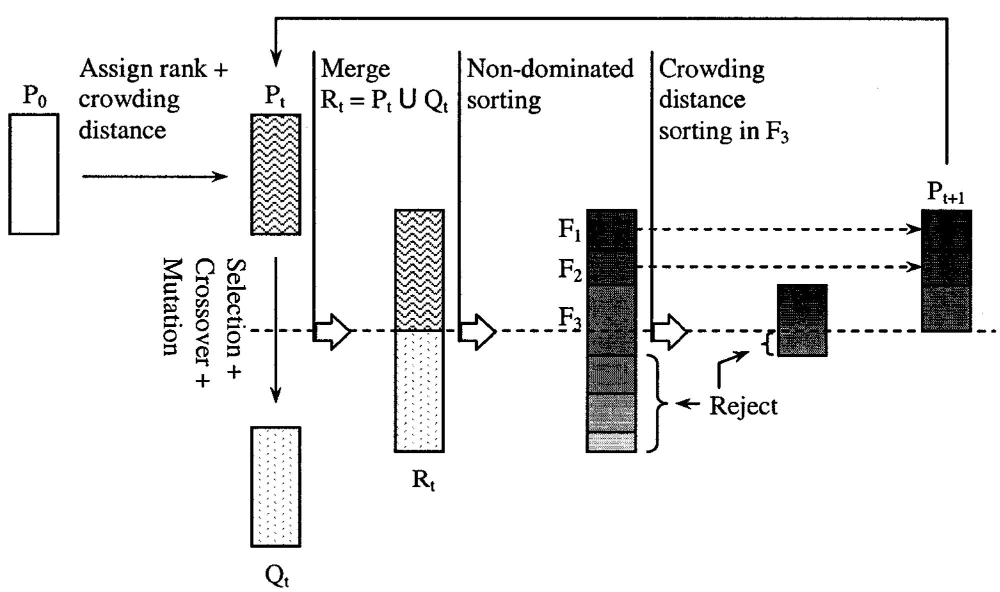



Figure 4.11 – NSGA-II Procedure

The crossover operator is maybe the most important element of the NSGA-II (of any genetic algorithm, actually). The real-coded NSGA-II uses the simulated binary crossover (SBX) operator introduced by Deb and Agrawal (1995). The crossover operator is controlled by two parameters, the probability of crossover, $P_{Cross}$, and the distribution index for crossover $\eta_c$. After two mating parents are selected through a tournament selection, a uniform random number $r_c$ in [0,1] is generated, and the crossover will take place if $r_c \leq P_{Cross}$. The distribution index $\eta_c$ is a very important and sensitive parameter which will define the effect of the crossover operator on the offspring population.

To demonstrate the importance of the distribution index, the following experiment was conducted. Fifty children were generated from the same two parents using the crossover operator with distribution index of 1, 5, 20, and 50. The resulting offspring populations can be seen in Figure 4.12. The spread of the offspring population increases considerably with decreasing values of the distribution index.

### 4.2.1. NSGA-II Solver Specific Features

The NSGA-II Solver was coded in VBA as close as possible to the code in C that the NSGA-II authors made publicly available. A few changes were necessary in order to accommodate differences in the two programming languages. The algorithm is essentially the original real-coded NSGA-II. The scripting tool for user-defined macros and the routine for dealing with equality constraints were included as extra features, making the NSGA-II Solver more general and flexible than the original code.

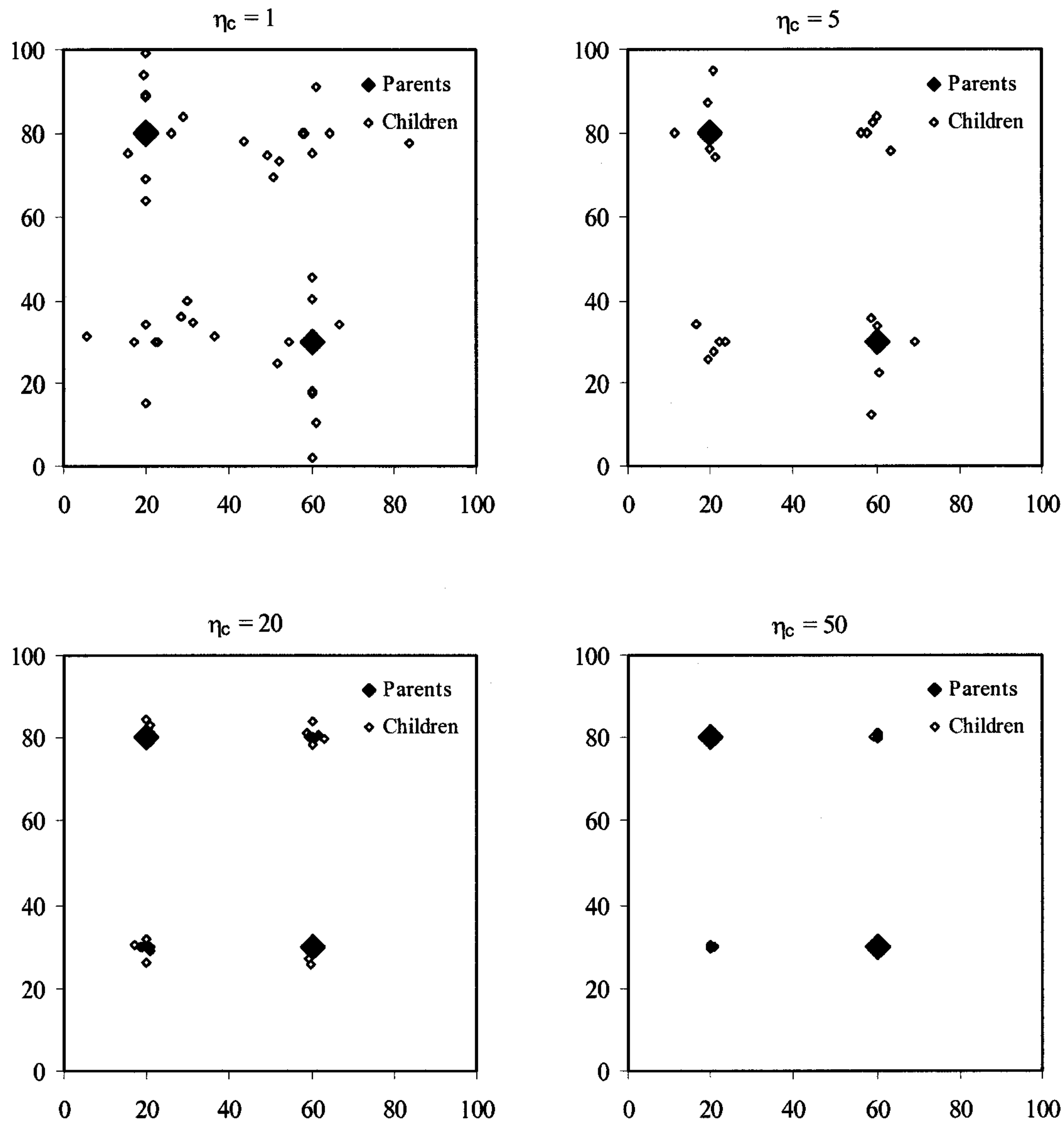


Figure 4.12 – Effect of the Distribution Index on the NSGA-II Crossover Operator

4.2.2. NSGA-II Parameters

The following parameters are adjustable in the NSGA-II Solver: (i) population size, (ii) number of generations, (iii) probability of crossover ($P_{cross}$), (iv) probability of mutation ($P_{mut}$), (v) distribution index for crossover ($\eta_c$), and (vi) distribution index for mutation ($\eta_m$).

The default parameters of the NSGA-II Solver are presented in Figure 4.13.

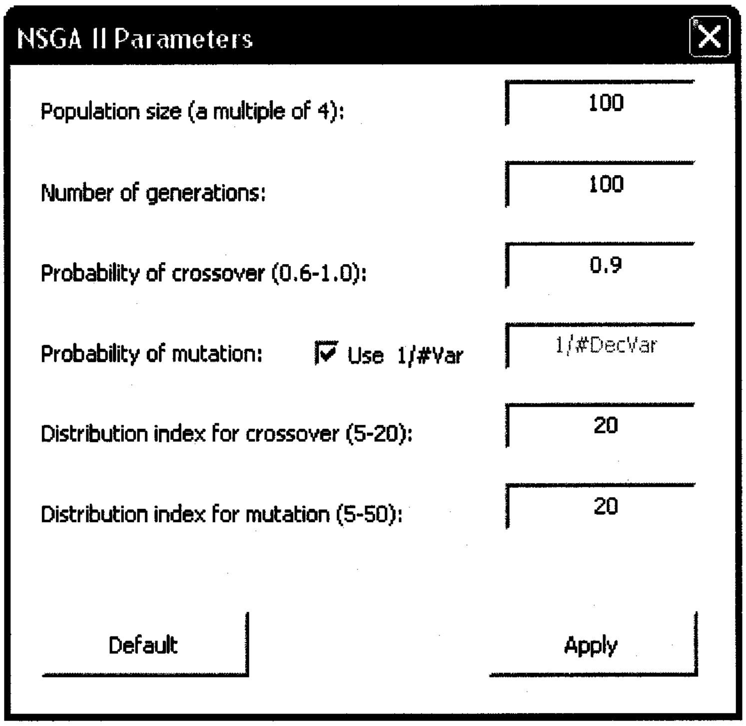



Figure 4.13 – NSGA-II Solver Default Parameters

### 4.3. ε-Constraint with Nonlinear Optimization

The ε-constraint method was described in section 2.2.1 of the literature review. The method was implemented as a generalized multi-objective solver for Excel. The ε-NLP Solver uses the Excel Solver® package as its nonlinear optimization engine. The key element of the ε-NLP Solver is the algorithm used to define the ε bounds.

The definition of the ε bounds is not a simple task, especially when the number of objectives increases. The best and worst optimal values for each objective have to be identified in order to set appropriate ε bounds. This is usually done by optimizing the objectives sequentially. The best values are easily obtained by optimizing each objective without any consideration to the others. The worst values are much more difficult to identify. The worst value for an objective will clearly correspond to a case in which that objective is last in the sequence to be optimized. The problem is that the values obtained for that objective will depend on the order in which the previous ones were optimized, and no one can know a priori which sequence will cause the worst value.

If the true worst values for each objective are not properly identified the solver will likely neglect part of the Pareto optimal front.

The ε-NLP Solver addresses this problem through a combinatorial approach. For each objective, all possible permutations of the remaining objectives are enumerated and the solver is run for all sequences in order to find the worst value for that objective. For a problem with three objectives, this will involve two sequences of optimizations for each of the two objectives represented as ε-constraints. For a problem with six objectives, 120 sequences of optimizations for each of the five ε-constraint objectives have to be performed. For more complex problems, for example with many discontinuities, highly nonlinear Pareto surfaces, or oddly constrained search spaces, the gradient-search based algorithm of the Excel Solver will easily fail to simply identify the ε bounds.

For most problems tested in this research, however, the ε-NLP Solver was successful in finding the whole extension of the Pareto optimal front; although in many cases a number of different initial solutions had to be used.

### 4.3.1. ε-NLP Solver Specific Features

After identifying the best and worst values for each ε-constraint objective, the ε values have to be parametrically varied within those limits while optimizing the primary objective. Traditionally, this was done by distributing linearly the ε values as a function of the desired number of solutions. Depending on the shape of the Pareto front, if it is highly nonlinear with a wide range of slopes, for example, this approach will produce a very uneven distribution of solutions along the Pareto front.

To overcome this issue, a special routine was developed to define the ε values sequentially taking into account information on the values of the primary objective, as a surrogate measure of derivatives. Figure 4.14 shows the pseudo-code for the routine.

```
f1 = objective value for the primary objective
εn = epsilon value for objective n
NrEps = number of values of εn to be set

Find f1(1) for εn(1) = εn^max
Find f1(3) for εn(3) = εn^min
RangeOb = ABS( f1(1) − f1(3))
RangeEps = ( εn^max − εn^min )
Find f1(2) for εn(2) = (εn^max + εn^min) / 2
L = 3
Do While L ≤ NrEps
  MaxDelta = 0
  {find the interval between two adjacent values of εn with the maximum slope, in either
  objective}
  For k = 1 to (L-1)
    DeltaOb = ABS[ f1(k) − f1(k+1)] / RangeOb
    DeltaEps = ABS[ εn(k) − εn(k+1)] / RangeEps
    If  (DeltaOb > MaxDelta) or (DeltaEps > MaxDelta) then
      MaxDelta = Max{DeltaOb, DeltaEps}
      NextEps = k
    Endif
  Next k
  {set next εn to the middle of the interval with maximum slope}
  εn(L+1) = (εn(NextEps) + εn(NextEps+1)) / 2
  L = L+1
Loop
```

Figure 4.14 – Pseudo-Code of the Slope-Based Routine for Epsilon Bounds

A comparison between the traditional and the proposed slope-based approaches can be found in Figure 4.15, using the SELECT model, with the same number of solutions.

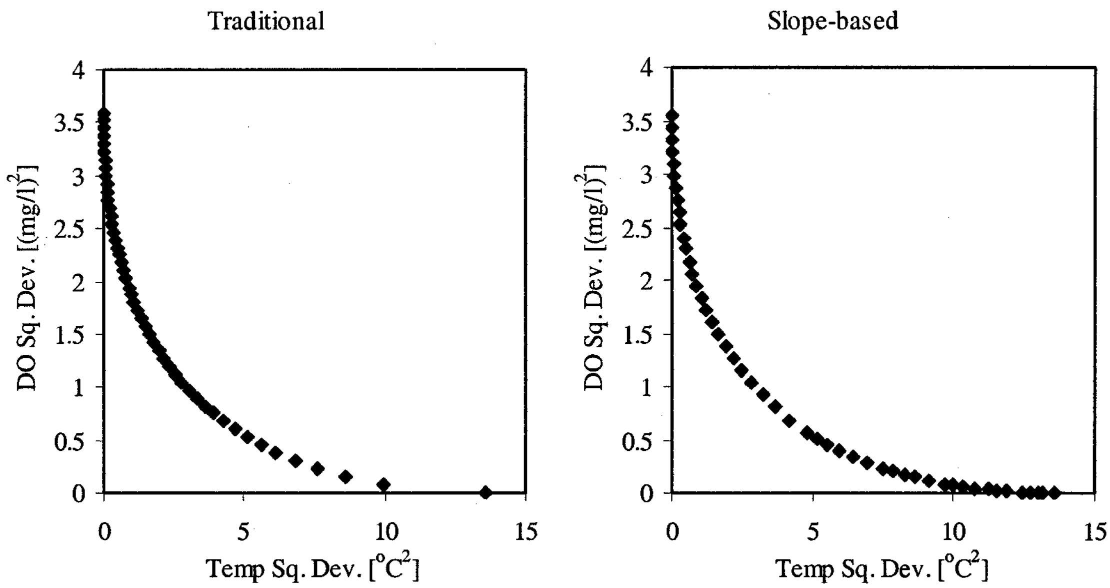



Figure 4.15 – Effect of the Slope-Based Routine for Epsilon Bounds

4.3.2. ε-NLP Solver Parameters

In addition to all parameters required by the Excel Solver, the ε-NLP Solver requires the definition of the number of desired solutions, and a feasibility tolerance that is used to independently check the feasibility of obtained solutions. This way the precision parameter of the Excel Solver can be kept at its small default value, forcing the Excel Solver to be more accurate. Some solutions would eventually be considered infeasible for that precision, but feasibility is checked after the solver is executed with the feasibility tolerance parameter, which may be less restrictive, and defined according to the actual precision the analyst judges appropriate for the constraints imposed to the problem. The default parameters of the e-NLP Solver are presented in Figure 4.16.

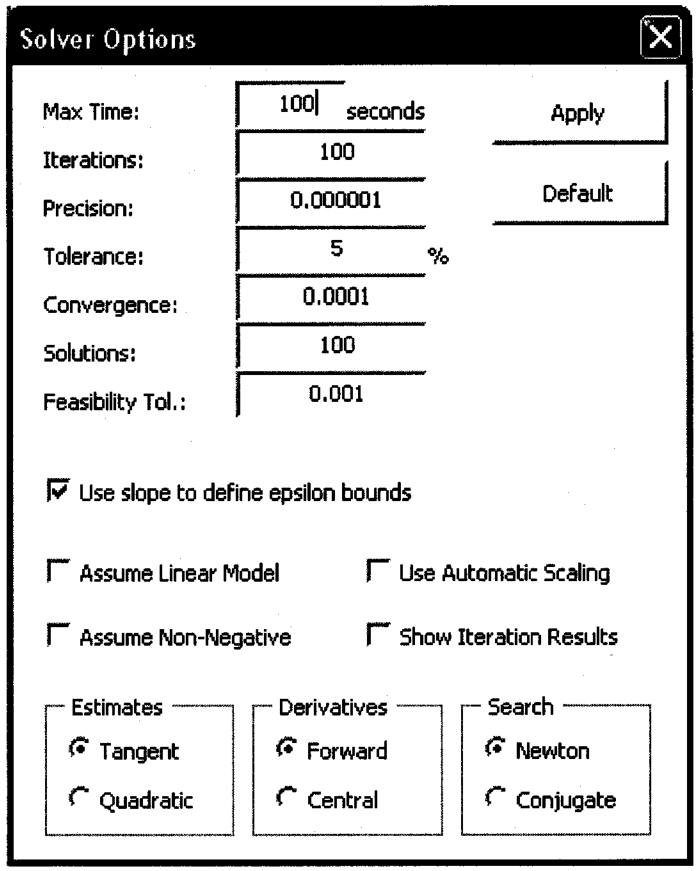



Figure 4.16 – ε-NLP Solver Default Parameters

## 4.4. Random Search Solver

The RSearch Solver tries to solve the multi-objective problem by simply generating a very large number of solutions randomly distributed throughout the bounded search space. The objective values are evaluated for each solution, and all solutions are compared to each other so that the non-dominated ones can be stored in an array that must be dynamically declared since the number of non-dominated solutions cannot be determined a priori.

The performance metrics used to compare the different methods usually involve computation of mean or variance measures, which are influenced by the size of the test set. All comparisons should then be made using test sets of approximately the same size.

The NSGA-II crowded-comparison approach was used in the RSearch Solver to select the requested number of solutions from the final set of non-dominated solutions.

The Solver also includes the user-defined macro scripting tool and the routine to handle equality constraints imposed on decision variables.

# 5. THE INTERACTIVE COMPROMISE COORDINATE METHOD

Problems in the field of multiple criteria decision-making (MCDM) are often divided into two types. In multi-objective optimization (MO), alternative solutions are not known a priori and are generated from the optimization. Usually an infinite number of Pareto optimal solutions exist. Alternatives can vary continuously and they are represented by decision variables restricted by a set of constraints. In multi-criteria decision analysis (MCDA), also called multi-attribute decision analysis, there exist a finite number of discrete feasible alternatives enumerated in advance. In both cases, however, in the presence of conflicting objectives or criteria, the decision maker's preference structure has to be modeled in order to rank, classify, or select best compromise alternatives. Furthermore, the adequate display of trade-offs between various alternatives can help decision makers understand and explore the entire range of potential solutions.

## 5.1. Methods for Displaying Results of Multi-Objective Analyses

The decision-making process is highly influenced by the way information is presented to the decision makers. This is valid for any kind of decision-making problem and especially true in MCDM problems, given the complexity introduced by the multiple dimensions. Mason and Mitroff (1973), in early research on information systems, highlighted the importance of the mode of presentation for effective man-machine communication. Dickson *et al.* (1977), synthesizing a series of experiments known as the "Minnesota experiments", suggested that graphic outputs can lead to faster and "better" decision making. Among other conclusions, these experiments also revealed the fact that managers like to use interactive systems.

With the development of computers and increasing graphical capabilities of object-oriented programming languages, the use of graphical illustration in MCDM problems was significantly enhanced. Miettinen (1999) and Deb (2001) provide good reviews of some graphical methods used in MCDM applications. Geoffrion et al. (1972) introduced the value path method, where each alternative is represented by a line which connects the values for each objective or criterion. All alternatives are shown in a two-dimensional plot where the objectives are equally distributed on the horizontal axis and the objective values are recorded on the vertical axis. The method also allows the display of the ranges for each objective.

Cleveland (1994) suggested the use of scatter-plot matrices to display multidimensional data. In multi-objective analysis, each panel in the matrix is a scatter-plot of two objective functions. Sobol and Klein (1989) used star graphs for financial analysis with different attributes. Each alternative is represented by a circle divided into *m* equal arcs, where *m* is the number of attributes, or objectives. The objectives are placed in the ending points of the arcs and their values are recorded in the radial lines that connect their respective points to the center of the circle. A polygon is formed by connecting the coordinates representing each objective value. The area of the polygon is used to rank the alternatives.

Tan and Fraser (1998) suggested a modified star graph that allows the weights to be displayed in the same plot. They showed that the order in which objectives are placed in the circle can change the area of the polygon and thus the final ranking. These authors also suggested the petal diagram method, which uses a circle as reference as well. The circle is divided in sectors each one representing one objective. The angular size of the sector represents the weight associated with the objective. In each sector a "petal" is drawn with the same angular size, with radial length reflecting the objective value. One circle with petals is used for each alternative. The star graph and the petal diagram, as well as the spider-web chart method discussed in Kasanen *et al.* (1991), have individual charts representing each alternative which makes these methods suitable for MCDA with relatively small number of alternatives but not appropriate for MO where there may exist hundreds of Pareto solutions to be explored. Losa *et al.* (2001) suggested the use of a covariance biplot technique resulting in a graphical output called a "conflict diagram". All alternatives are presented in a single diagram. However, interpretation of biplot graphs is not straightforward and decision makers would likely need some experience to use these techniques (Miettinen, 1999).

This dissertation introduces a new method called Interactive Compromise Coordinate (ICC). The ICC method is particularly suitable for MO problems where a large number of alternatives exist but the method can also be used for MCDA problems. The ICC method allows the DM to explore MO Pareto sets and MCDA results no matter what approach was used to generate the Pareto solutions or to rank the MCDA alternatives. The only assumption is that the DM's preference structure may be modeled by a set of weights, which is the case in most MCDM methods.

### 5.2. The Interactive Compromise Coordinate Method

A method to visualize and explore solutions of multi-objective optimization and multi-criteria decision analysis was introduced. The Interactive Compromise Coordinate (ICC) method allows the visualization of all alternative solutions in a single graph. The decision maker can explore MO Pareto optimal sets and MCDA results no matter what method was used to rank the alternatives. The only assumptions are that the DM's preference structure can be modeled by a set of weights and that all alternatives are transitively comparable to each other, i.e. a complete pre-order is obtainable. The ICC method was implemented in VBA as an add-in to Excel and also included in the multi-objective solvers developed in this research.

The user can rank alternative solutions using any method programmed into the spreadsheet and display the results using the ICC stand-alone Excel add-in. If no specific MCDA method is used, the ICC method employs a compromise programming approach with an Euclidian norm (L2-norm) to rank the alternatives based upon the set of weights. The weights can be changed interactively and the user can immediately observe new best compromise solutions, and different groups of solutions with similar performance over all objectives. The ICC add-in allows the consideration of up to eight objectives.

The ICC method uses a single graph to represent all alternative solutions. First, all Pareto solutions, or the MCDA pay-off matrix, are normalized using an absolute norm (L-1 norm). All objectives are placed as vertices equally spaced in a circumference of diameter *one*. Let $P$ be the number of objectives. The first objective is arbitrarily assigned to coordinates [0.5,1.0]. The $x$ and $y$ coordinates of the following objectives are given by the following equations:

$$\theta = \frac{\left(\pi - \frac{2\pi}{P}\right)}{2} \tag{5.1}$$

$$x_{ob}(k) = x_{ob}(k-1) + Cos\theta \cdot Cos\left[\frac{-\pi}{2} + \pi \cdot k - (2 \cdot k - 3) \cdot \theta\right] \tag{5.2}$$

$$y_{ob}(k) = y_{ob}(k-1) + Cos\theta \cdot Sin\left[\frac{-\pi}{2} + \pi \cdot k - (2 \cdot k - 3) \cdot \theta\right] \tag{5.3}$$

where: $P$ is the number of objectives and $k = 2,3,\ldots,P$.

The normalized metrics are calculated for each objective as follows:

$$CP(i,k) = \frac{Best(k) - R(i,k)}{Best(k) - Worst(k)} \tag{5.4}$$

where: $CP(i,k)$ is the [0,1] normalized metric of alternative $i$ for objective $k$, $Best(k)$ is the best value for objective $k$ over all alternatives, $Worst(k)$ is the worst value of objective $k$ over all alternatives, and $R(i,k)$ is the value of objective $k$ for alternative $i$. Again, in the case of MO, the alternatives are defined in the decision (or parameter) space, by the values of the decision variables, and mapped to a solution vector in the objective space, whose components are given by $R(i,k)$ in Eq. 5.4. The $(x_{ob,} y_{ob})$ points in the ICC graph represent the *ideal objective vector* formed by the best values of each objective individually. The points diametrically opposed to $(x_{ob,} y_{ob})$ represent the *nadir objective vector*, corresponding to the worst values of each objective individually.

The normalized metric for each objective is used to obtain a new coordinate measured in the diameter corresponding to that objective. The new coordinates are given by:

$$x_{CP}(i,k) = x_{ob}(k) + CP(i,k) \cdot Cos\left[\frac{\pi}{2} + \pi \cdot k - (2 \cdot k - 2) \cdot \theta\right] \tag{5.5}$$

$$y_{CP}(i,k) = y_{ob}(k) + CP(i,k) \cdot Sin\left[\frac{\pi}{2} + \pi \cdot k - (2 \cdot k - 2) \cdot \theta\right] \tag{5.6}$$

For example, if an alternative has a CP metric of zero for an objective, i.e. it has the best value for that objective, the $x_{CP}$ and $y_{CP}$ coordinates for that objective will coincide with the $x_{ob}$ and $y_{ob}$ coordinates, exactly on the vertex corresponding to that objective. Conversely, a CP metric of one would imply the farthest coordinate (diametrically opposed) for a particular objective.

Each alternative has a $(x_{CP}, y_{CP})$ coordinate for each of the $P$ objectives. The alternative is then plotted in the centroid defined by these $P$ points using a marker. The positions of the MO Pareto solutions or MCDA alternatives in the ICC graph are independent of the weights and remain unchanged when the weights are varied. The marker's size and color, however, will depend on the overall performance of the alternative under all objectives, which is obtained as a function of the weights assigned to all objectives. When applying a specific MCDA method, one has to provide the cell (range) references in the spreadsheet where the overall performance metrics for all alternatives and the weights for all objectives are located. This operation is done using the input form (Figure 5.1) that is loaded when the ICC method is called from the Tools menu in Excel. If no particular MCDA method is specified, the L-2 compromise programming method is automatically used. The ICC method interface is shown in Figure 5.2. The weights can be changed in the interface and the graph will be immediately updated.

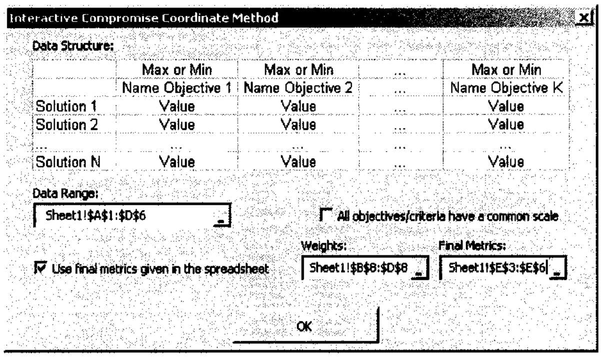



Figure 5.1 – ICC Input Form

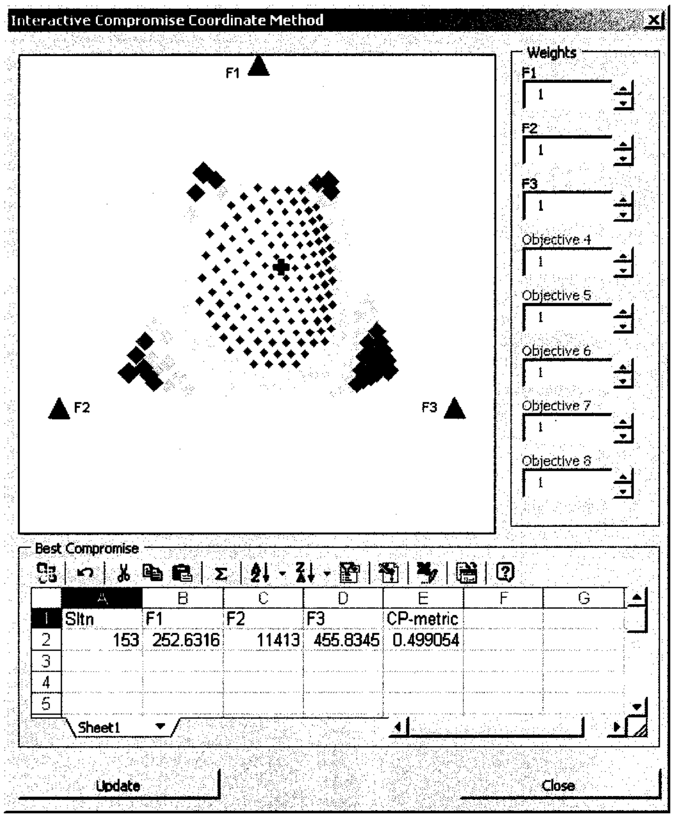



Figure 5.2 – ICC Interface

The alternative marked with a cross in Figure 5.2 is the best compromise. Larger markers correspond to worse alternatives. The color convention goes from light green (best) to dark red (worst), with a total of seven categories.

The projection effect of the ICC method may be visualized in Figure 5.3 where hypothetical three-dimensional Pareto surfaces are represented.

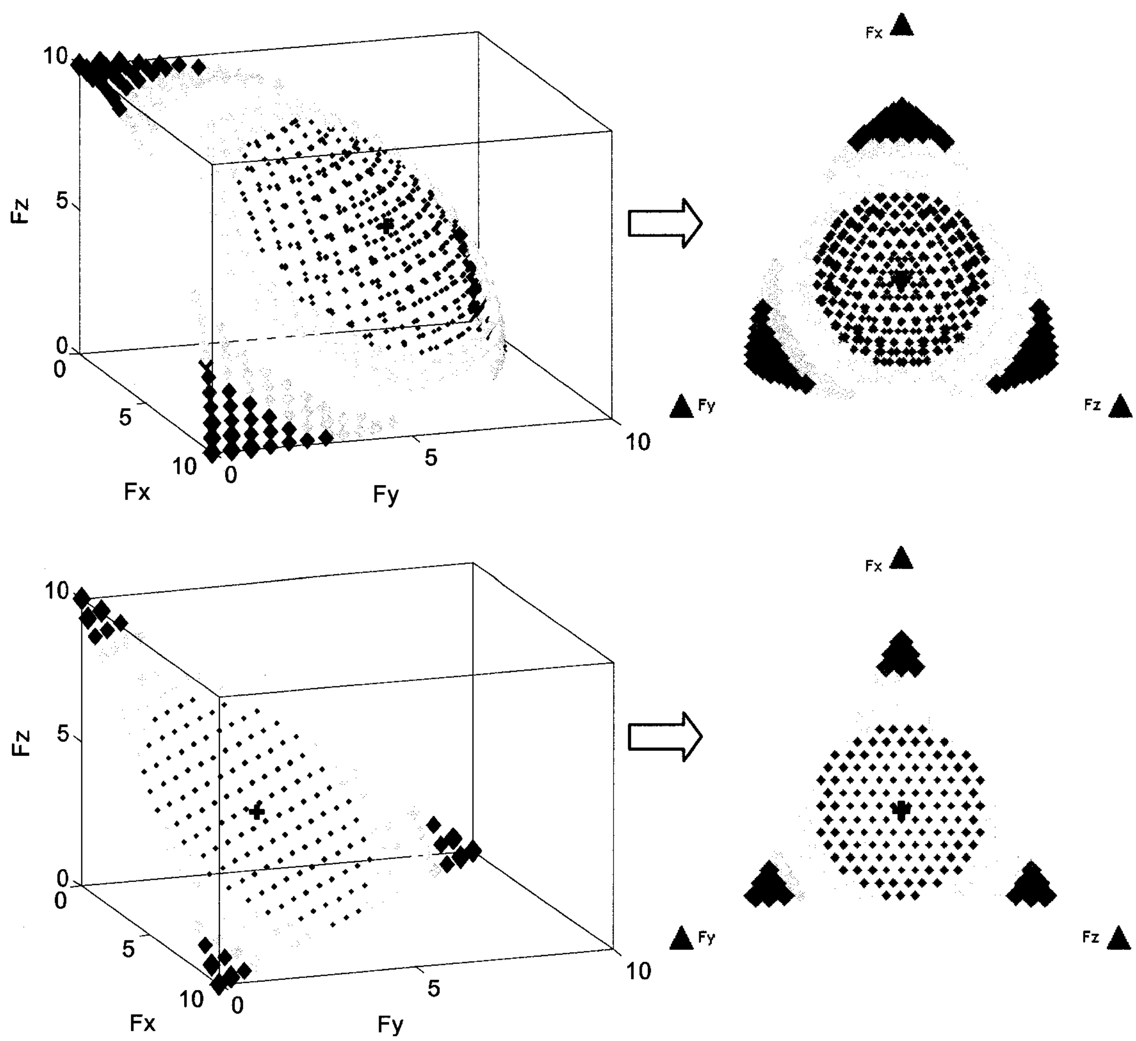



Figure 5.3 – ICC Visualization of Known 3-D Surfaces

In a MO problem with two objectives the ICC representation of the Pareto curve is a single line as shown in Figure 5.4.

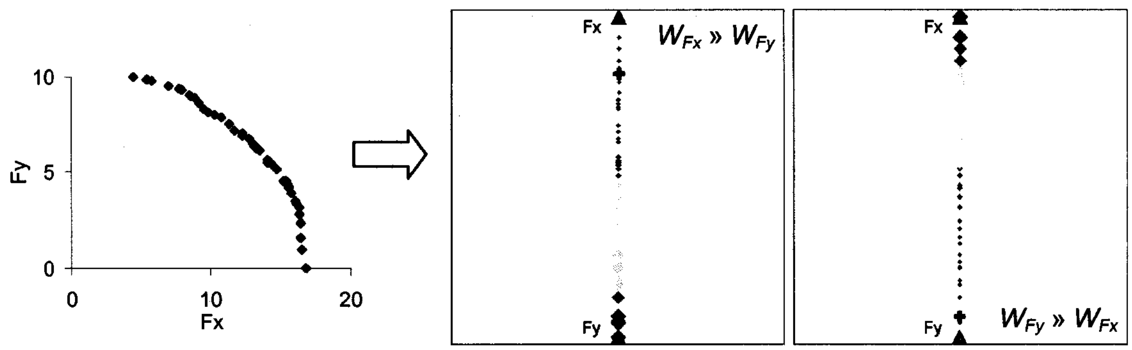



Figure 5.4 – ICC Graph for a 2-D Pareto Curve for Different Weights

The ICC interface allows the user to select, directly from the graph, any individual alternative or group of alternatives in the same compromise class and automatically export them to the table (which is actually an independent worksheet object) shown in Figure 5.2. Specific solutions can be easily copied from the ICC interface spreadsheet to other documents or software for further analysis.

The ICC method was used in the analysis of all multi-objective water resources problems presented in Chapter 6, involving two, three, four, and six objectives. Applications of the ICC method to MCDA problems with seven and eight objectives can be found in Baltar and Fontane (2006).

# 6. APPLICATIONS

The results of the multi-objective optimizations of the selected test functions and water resources problems are presented and discussed in this chapter. The different multi-objective solvers were run thirty times and statistics of the several performance metrics are compared. The performance metrics for all runs are presented in Appendix A. For the test functions, results obtained by other authors are also presented. The same number of function evaluations was always employed for the MOPSO and NSGA-II Solvers in order to produce a fair comparison. For the RSearch Solver, a larger number of function evaluations was used, usually thirty thousand, since the method is much simpler.

## 6.1. Test Functions

For all test functions (unless specified), the MOPSO and NSGA-II Solvers were run with the same parameters as reported in the corresponding reference used for comparison and validation. In the result tables to follow, MOPSO-e refers to the MOPSO Solver with crossover on extremes, and MOPSO-p refers to the MOPSO with post-processing. In each table, the best average values are always highlighted in boldface.

### 6.1.1. Test Problem 1

This multi-objective problem was proposed by Kita et al. (1996) and it was used by Coello Coello et al. (2004). It has two objective functions and two decision variables, constrained by three linear inequalities. The efficient solutions lie on the boundary of the first constraint. The feasible region and the Pareto optimal set are shown in Figure 6.1.

Maximize $f_1(x_1, x_2)$, (6.1)

Maximize $f_2(x_1, x_2)$

where:

$$f_1(x_1, x_2) = -x_1^2 + x_2, \qquad f_2(x, y) = \frac{x_1}{2} + x_2 + 1$$

subject to:

$$\frac{x_1}{6} + x_2 - 6.5 \leq 0, \quad \frac{x_1}{2} + x_2 - 7.5 \leq 0, \quad 5x_1 + x_2 - 30 \leq 0, \quad 0 \leq x_1, x_2 \leq 7$$

The same test function was also used by Gill et al. (2006) for comparison with their implementation of MOPSO.

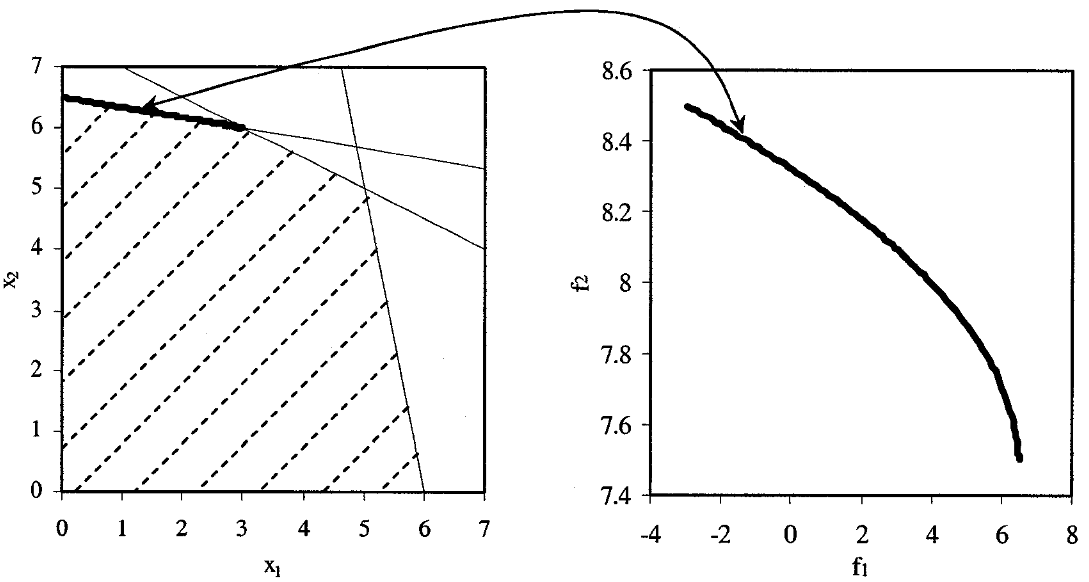



Figure 6.1 – Feasible Region and Pareto Optimal Front for Test Problem 1

This problem was solved using a population size of 100 and 50 cycles, or generations in NSGA terminology. The same number of 5000 function evaluations was used by Coello Coello et al. (2004). The results are presented in Tables 6.1 to 6.9 and Figure 6.2.

Table 6.1 – Generational Distance Metric for Test Problem 1

| Problem 1 | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 0.003143 | 0.003337 | 0.002949 | 0.003242 | 0.005023 |
| Worst | 0.137047 | 0.149858 | 0.115966 | 0.159072 | 0.375358 |
| Average | 0.018531 | **0.017457** | 0.017893 | 0.030687 | 0.176296 |
| Median | 0.005995 | 0.005416 | 0.011336 | 0.013646 | 0.171419 |
| Std.Dev. | 0.034282 | 0.031346 | 0.022977 | 0.036568 | 0.115240 |

Table 6.2 – GD Reference Values for Test Problem 1

| Problem 1 | Coello Coello et al. (2004) | | Gill et al. (2006) |
|---|---|---|---|
| | MOPSO | NSGA-II | MOPSO |
| Best | 0.002425 | 0.003885 | 0.0122* |
| Worst | 0.476815 | 0.678449 | |
| Average | 0.036535 | 0.084239 | |
| Median | 0.007853 | 0.011187 | |
| Std.Dev. | 0.104589 | 0.165244 | |

These authors reported the results of only one run of their algorithm.

Table 6.3 – Spacing Metric for Test Problem 1

| Problem 1 | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 0.040680 | 0.039822 | 0.042139 | 0.035908 | 0.122810 |
| Worst | 1.370109 | 1.498565 | 1.062328 | 1.038797 | 1.920278 |
| Average | 0.174463 | **0.141844** | 0.144603 | 0.178657 | 0.695544 |
| Median | 0.064860 | 0.055924 | 0.085357 | 0.049715 | 0.601548 |
| Std.Dev. | 0.324635 | 0.277241 | 0.197858 | 0.247334 | 0.527502 |

Table 6.4 – Spacing Reference Values for Test Problem 1

| Problem 1 | Coello Coello et al. (2004) | | Gill et al. (2006) |
|---|---|---|---|
| | MOPSO | NSGA-II | MOPSO |
| Best | 0.043982 | 0.001032 | |
| Worst | 0.538102 | 1.488680 | |
| Average | 0.109452 | 0.098486 | 0.1415* |
| Median | 0.067480 | 0.027173 | |
| Std.Dev. | 0.110051 | 0.327380 | |

These authors reported the results of only one run of their algorithm.

Table 6.5 – Inverted Generational Distance Metric for Test Problem 1

| Problem 1 | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 0.004809 | 0.004836 | 0.004570 | 0.005507 | 0.008950 |
| Worst | 0.006434 | 0.011007 | 0.006730 | 0.012773 | 0.014914 |
| Average | 0.005550 | 0.005779 | **0.005447** | 0.007669 | 0.011663 |
| Median | 0.005644 | 0.005491 | 0.005400 | 0.006843 | 0.011480 |
| Std.Dev. | 0.000386 | 0.001077 | 0.000521 | 0.001934 | 0.001481 |

Table 6.6 – Diversity Metric for Test Problem 1

| Problem 1 | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 0.595958 | 0.612450 | 0.561810 | 0.620062 | 0.698981 |
| Worst | 1.132237 | 1.132643 | 1.060438 | 1.134517 | 1.139186 |
| Average | 0.725388 | 0.735889 | **0.725061** | 0.797626 | 0.945803 |
| Median | 0.683959 | 0.705697 | 0.711521 | 0.763189 | 0.940574 |
| Std.Dev. | 0.132209 | 0.109601 | 0.098850 | 0.144896 | 0.130018 |

Table 6.7 – Processing Time for Test Problem 1

| Problem 1 | Excel Add-ins (sec) | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 4.56 | 7.12 | 5.59 | 5.72 | 11.02 |
| Worst | 8.02 | 9.39 | 7.41 | 5.83 | 12.17 |
| Average | 6.07 | 8.31 | 6.55 | **5.75** | 11.43 |
| Median | 6.04 | 8.30 | 6.52 | 5.75 | 11.41 |
| Std.Dev. | 0.62 | 0.54 | 0.41 | 0.03 | 0.24 |

Table 6.8 – Dominated Ratio for Test Problem 1

| Problem 1 | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 0.45 | 0.46 | 0.20 | 0.48 | 0.73 |
| Worst | 0.74 | 0.68 | 0.42 | 0.63 | 0.94 |
| Average | 0.58 | 0.57 | **0.31** | 0.57 | 0.84 |
| Median | 0.58 | 0.58 | 0.31 | 0.58 | 0.84 |
| Std.Dev. | 0.07 | 0.06 | 0.05 | 0.05 | 0.06 |

Table 6.9 – Dominated Degree for Test Problem 1

| Problem 1 | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 0.47% | 0.44% | 0.26% | 0.60% | 1.13% |
| Worst | 1.88% | 1.90% | 2.60% | 2.77% | 6.90% |
| Average | 0.75% | 0.73% | **0.62%** | 1.00% | 3.00% |
| Median | 0.64% | 0.63% | 0.45% | 0.86% | 2.79% |
| Std.Dev. | 0.32% | 0.35% | 0.49% | 0.46% | 1.30% |

*Analysis and Comparison*

The results of the MOPSO and NSGA-II Solvers were very similar to those reported by Coello Coello et al. (2004), slightly better in terms of convergence (GD) and slightly worse in terms of spacing. In both cases, however, the standard deviations are relatively large compared to the means (coefficients of variation larger than 1.5), denoting a rather variable performance on these metrics. The large variation is also observed in the reference values by Coello Coello et al (2004). On the inverted GD and diversity metrics the MOPSO, MOPSO-e, and MOPSO-p had very similar performance, followed closely by the NSGA-II. The coefficients of variation for these metrics are much smaller. The iGD and diversity results indicate that all algorithms, including the NSGA-II, usually cover the whole extension of the Pareto optimal set but keep non-optimal solutions on the extremes in a number of runs (this can also be seem in Figure 6.2). This behavior also explains the large coefficients of variation for the GD and spacing metrics, where the extreme non-optimal solutions generate outlier values that raise the average values for those metrics (the medians are much smaller).

While the dominated ratio shows that about 57 percent of the solutions for MOPSO, MOPSO-e, and NSGA-II and 31 percent for MOPSO-p are dominated, the small dominated degrees (less than one percent) show that the solutions are generally very close to optimal.

The performance of the pure random search is generally much worse in all metrics.

### 6.1.2. Test Problem 2

This is a multi-modal, unconstrained multi-objective problem proposed by Deb (1999), which was used by Coello Coello et al. (2004). A bimodal two-objective optimization problem is created by using the bimodal function $g(x_2)$ shown in Figure 6.3. The problem has a local Pareto optimal front corresponding to $x_2 \approx 0.6$ and a global Pareto optimal front at $x_2 \approx 0.2$, which are both shown in Figure 6.3.

$$\text{Minimize } f_1(x_1, x_2), \qquad (6.2)$$

$$\text{Minimize } f_2(x_1, x_2)$$

where:

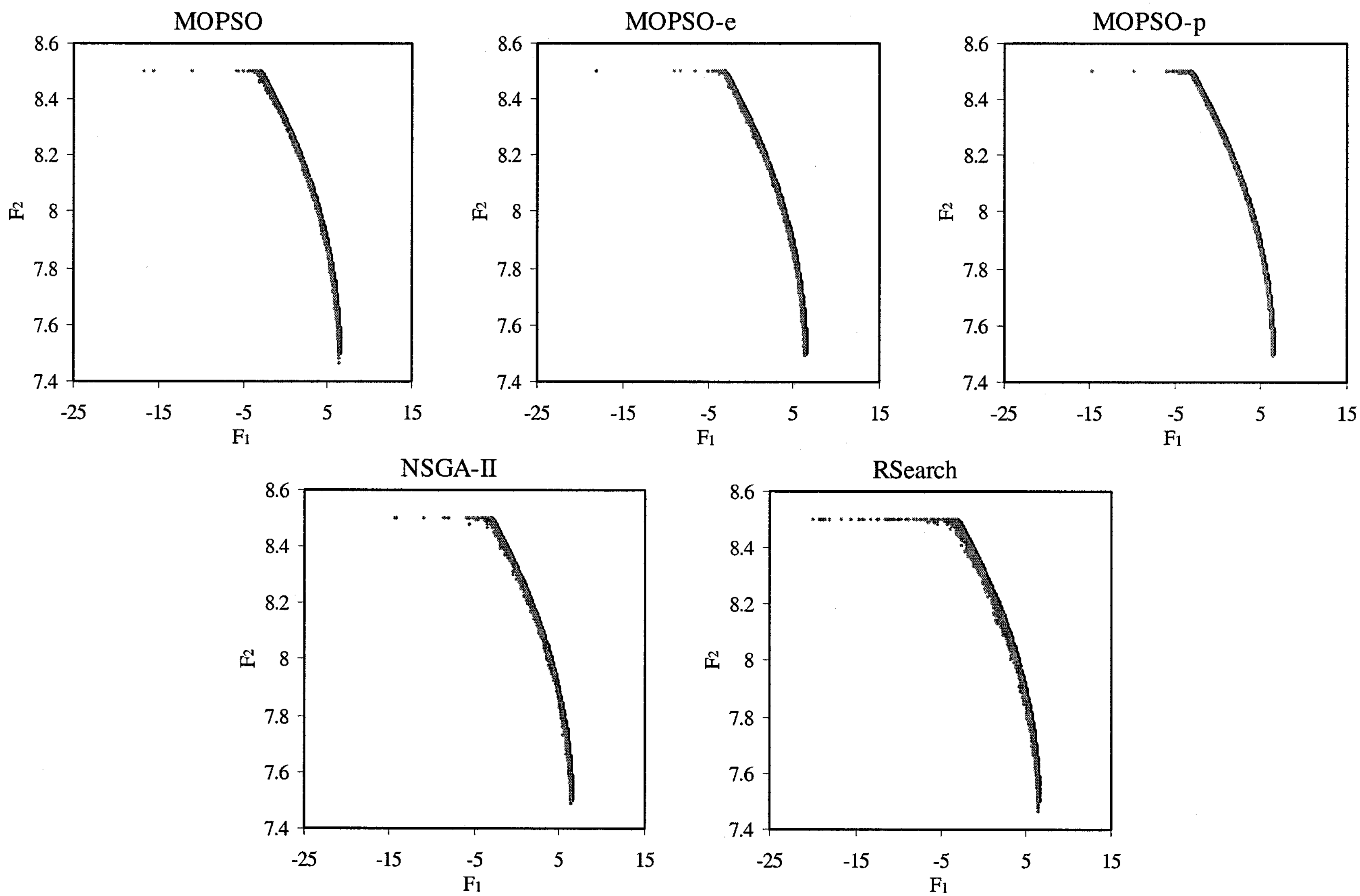
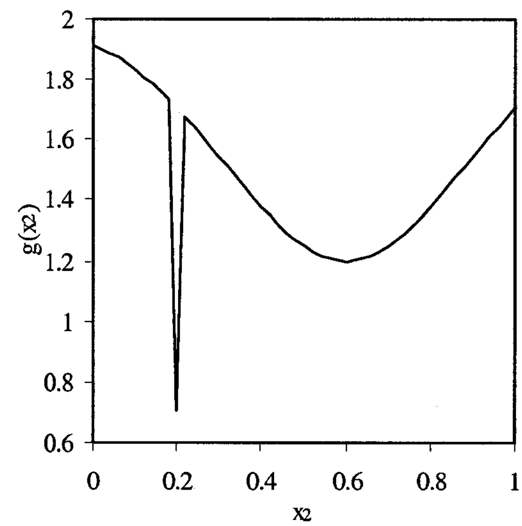


Figure 6.2 – Results for Test Problem 1

$$f_1(x_1, x_2) = x_1, \qquad f_2(x_1, x_2) = \frac{g(x_2)}{x_1}$$

$$g(x_2) = 2.0 - \exp\left[-\left(\frac{x_2 - 0.2}{0.004}\right)^2\right] - 0.8 \cdot \exp\left[-\left(\frac{x_2 - 0.6}{0.4}\right)^2\right]$$

$$0.1 \leq x_1, x_2 \leq 1.0$$



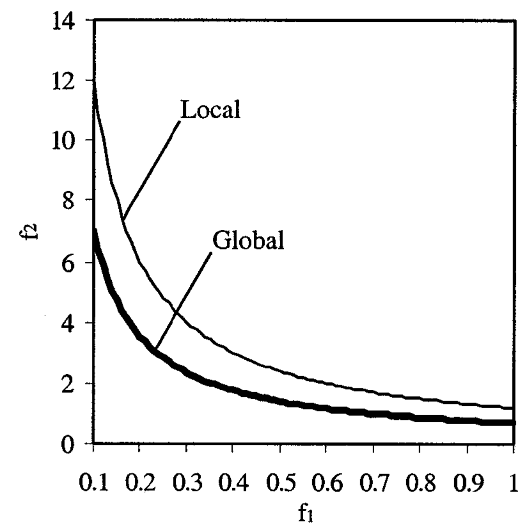




Figure 6.3 – Local and Global Pareto Optimal Fronts for Test Problem 2

This problem was solved using a population size of 100 and 100 generations. The same number of 10000 function evaluations was used by Coello Coello et al. (2004). The results are presented in Tables 6.10 to 6.18 and Figure 6.4.

Table 6.10 – Generational Distance Metric for Test Problem 2

| Problem 2 | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 0.006670 | 0.006430 | 0.006832 | 0.006651 | 0.208477 |
| Worst | 0.270149 | 0.202686 | 0.179426 | 0.161663 | 0.367636 |
| Average | 0.069281 | 0.047391 | 0.061984 | **0.044581** | 0.297776 |
| Median | 0.008366 | 0.008045 | 0.008280 | 0.007695 | 0.294691 |
| Std.Dev. | 0.084279 | 0.073138 | 0.072631 | 0.058234 | 0.042362 |

Table 6.11 – Generational Distance Reference Values for Test Problem 2

| Problem 2 | Coello Coello et al. (2004) | |
|---|---|---|
| | MOPSO | NSGA-II |
| Best | 0.000430 | 0.000700 |
| Worst | 0.185310 | 0.208467 |
| Average | 0.032730 | 0.044236 |
| Median | 0.000510 | 0.000856 |
| Std.Dev. | 0.060620 | 0.073680 |

Table 6.12 – Spacing Metric for Test Problem 2

| Problem 2 | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 0.028250 | 0.028308 | 0.034550 | 0.027859 | 0.077121 |
| Worst | 0.156542 | 0.095670 | 0.082946 | 0.964998 | 0.643191 |
| Average | 0.048448 | **0.044437** | 0.048822 | 0.110449 | 0.299220 |
| Median | 0.045501 | 0.041394 | 0.041669 | 0.036860 | 0.255398 |
| Std.Dev. | 0.023442 | 0.013763 | 0.013924 | 0.221534 | 0.163218 |

Table 6.13 – Spacing Reference Values for Test Problem 2

| Problem 2 | Coello Coello et al. (2004) | |
|---|---|---|
| | MOPSO | NSGA-II |
| Best | 0.040070 | 0.026086 |
| Worst | 0.581850 | 0.061422 |
| Average | 0.083580 | 0.037447 |
| Median | 0.054940 | 0.035529 |
| Std.Dev. | 0.118210 | 0.009238 |

Table 6.14 – Inverted Generational Distance Metric for Test Problem 2

| Problem 2 | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 0.003723 | 0.003385 | 0.003342 | 0.001699 | 0.007089 |
| Worst | 0.061793 | 0.129372 | 0.072760 | 0.033506 | 0.017429 |
| Average | 0.017957 | 0.022228 | 0.018865 | **0.007472** | 0.010727 |
| Median | 0.006558 | 0.006252 | 0.006782 | 0.002290 | 0.010037 |
| Std.Dev. | 0.017021 | 0.031937 | 0.018932 | 0.011832 | 0.002458 |

Table 6.15 – Diversity Metric for Test Problem 2

| Problem 2 | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 0.687735 | 0.670272 | 0.693122 | 0.539223 | 0.996884 |
| Worst | 1.043060 | 0.911658 | 0.899281 | 1.169907 | 1.209031 |
| Average | 0.786014 | 0.775076 | 0.787446 | **0.698418** | 1.133662 |
| Median | 0.790650 | 0.771183 | 0.790639 | 0.604915 | 1.143357 |
| Std.Dev. | 0.066072 | 0.064595 | 0.059268 | 0.193396 | 0.048007 |

Table 6.16 – Processing Time for Test Problem 2

| Problem 2 | Excel Add-ins (sec) | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 34.16 | 41.44 | 62.19 | 13.73 | 9.97 |
| Worst | 129.48 | 138.45 | 128.16 | 14.36 | 12.66 |
| Average | 90.92 | 99.18 | 93.08 | 13.83 | **11.18** |
| Median | 88.61 | 99.18 | 88.25 | 13.80 | 11.17 |
| Std.Dev. | 21.79 | 20.02 | 18.34 | 0.11 | 0.62 |

Table 6.17 – Dominated Ratio for Test Problem 2

| Problem 2 | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 0.00 | 0.00 | 0.01 | 0.01 | 0.50 |
| Worst | 1.00 | 1.00 | 1.00 | 1.00 | 0.85 |
| Average | 0.37 | 0.25 | 0.38 | **0.20** | 0.68 |
| Median | 0.02 | 0.02 | 0.02 | 0.04 | 0.68 |
| Std.Dev. | 0.48 | 0.42 | 0.48 | 0.37 | 0.07 |

Table 6.18 – Dominated Degree for Test Problem 2

| Problem 2 | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 0.00% | 0.00% | 0.00% | 0.00% | 6.76% |
| Worst | 26.72% | 24.10% | 21.05% | 26.55% | 23.10% |
| Average | 7.71% | **5.03%** | 7.34% | 5.91% | 13.56% |
| Median | 0.00% | 0.00% | 0.00% | 0.04% | 13.81% |
| Std.Dev. | 10.37% | 9.28% | 9.82% | 9.07% | 4.44% |

*Analysis and Comparison*

The performance of MOPSO and NSGA-II are very similar, with NSGA-II slightly better in all metrics but spacing. The MOPSO-e performed better than MOPSO and MOPSO-p and very close to NSGA-II. The results obtained by the MOPSO and NSGA-II solvers are comparable to those reported by Coello Coello et al. (2004) for GD and spacing, worse for GD and better for spacing, but with much smaller coefficients of variation in both cases.

The convergence metrics (GD, iGD) show higher variations than the diversity or distribution metrics (SP, DI). The dominated ratio varied from 0.20 for NSGA-II to 0.38 for MOPSO-p while the dominated degrees went from about 5% for MOPSO-e to about 8% for MOPSO.

The high values of the dominated ratio and degrees, and the different variations between convergence and diversity metrics as well, are due to the fact that all methods converged to the local optimal front in certain number of runs. MOPSO converged to the local front in eleven of the thirty runs, while NSGA-II converged to the local front in five of the thirty runs but kept highly non-optimal solutions on extremes in other five runs (which did not happen with MOPSO). MOPSO processing times were considerably higher than those for NSGA-II.

The pure random search presented the worst performance metrics overall but, as one can see from Figure 6.4, is not attracted to the local optimal front. This behavior is reasonable since the search is purely random and a local optimal solution found early does not influence the ability of the algorithm to find another one closer to the global optimal.

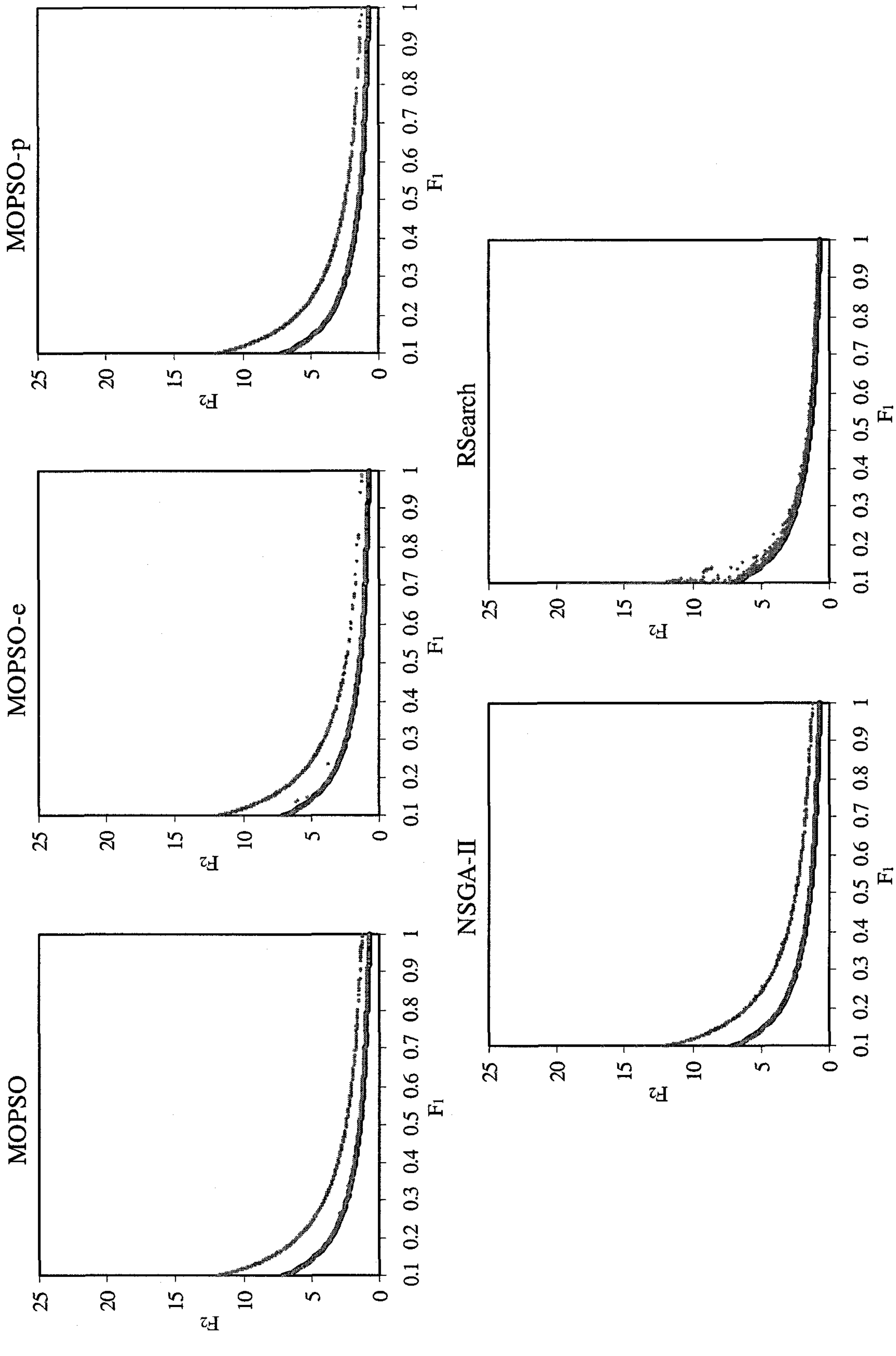



Figure 6.4 – Results for Test Problem 2

6.1.3. Test Problem 3

This two-objective problem was originally proposed by Kursawe (1991) and used by Coello Coello et al. (2004) and by Deb et al. (2002). The problem has three decision variables and a non-convex, disconnected Pareto optimal set. Many MOEA algorithms have trouble keeping a good spread of non-dominated solutions and often miss at least one of the disconnected portions of the Pareto optimal front shown in Figure 6.5.

Minimize $f_1(x_1, x_2, x_3)$, (6.3)

Minimize $f_2(x_1, x_2, x_3)$

where:

$$f_1(x_1, x_2, x_3) = \sum_{i=1}^{2}\left[-10 \cdot \exp\left(-0.2\sqrt{x_i^2 + x_{i+1}^2}\right)\right]$$

$$f_2(x_1, x_2, x_3) = \sum_{i=1}^{3}\left[|x_i|^{0.8} + 5\sin\left(x_i^3\right)\right]$$

$-5 \leq x_1, x_2, x_3 \leq 5$

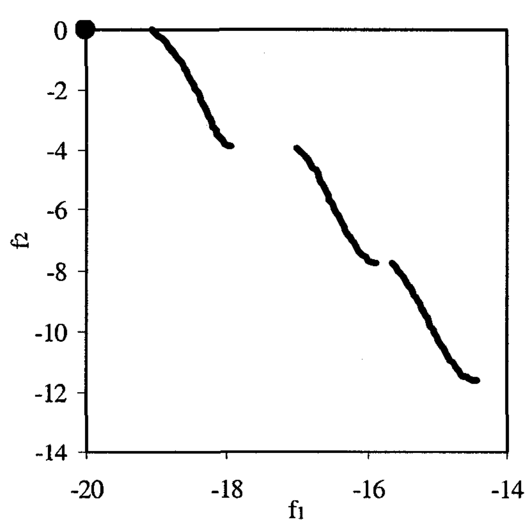



Figure 6.5 – Pareto Optimal Set for Test Problem 3

This problem was solved using a population size of 100 and 120 generations. The same number of 12000 function evaluations was used by Coello Coello et al. (2004). Deb et al. (2002) used 25000 function evaluations. The results are presented in Tables 6.19 to 6.28 and Figure 6.6.

Table 6.19 – Generational Distance Metric for Test Problem 3

| Problem 3 | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 0.004436 | 0.004276 | 0.004400 | 0.003564 | 0.077429 |
| Worst | 0.007370 | 0.007423 | 0.007830 | 0.004215 | 0.190253 |
| Average | 0.005696 | 0.005319 | 0.005505 | **0.003967** | 0.111559 |
| Median | 0.005648 | 0.005029 | 0.005152 | 0.003967 | 0.100196 |
| Std.Dev. | 0.000721 | 0.000811 | 0.000952 | 0.000167 | 0.030396 |

Table 6.20 – Generational Distance Reference Values for Test Problem 3

| Problem 3 | Coello Coello et al. (2004) | | Deb et al. (2002) |
|---|---|---|---|
| | MOPSO | NSGA-II | NSGA-II |
| Best | 0.007450 | 0.006905 | |
| Worst | 0.009600 | 0.103095 | |
| Average | 0.008450 | 0.029255 | 0.028964 |
| Median | 0.008450 | 0.017357 | |
| Std.Dev. | 0.000510 | 0.027170 | 0.004243 |

Table 6.21 – Spacing Metric for Test Problem 3

| Problem 3 | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 0.064576 | 0.056757 | 0.065539 | 0.056072 | 0.262144 |
| Worst | 0.164352 | 0.207753 | 0.168407 | 0.109915 | 0.837996 |
| Average | 0.111815 | 0.126849 | 0.103163 | **0.095875** | 0.484115 |
| Median | 0.114792 | 0.130526 | 0.103384 | 0.099384 | 0.469158 |
| Std.Dev. | 0.019747 | 0.027541 | 0.025010 | 0.012528 | 0.149809 |

Table 6.22 – Spacing Reference Values for Test Problem 3

| Problem 3 | Coello Coello et al. (2004) | |
|---|---|---|
| | MOPSO | NSGA-II |
| Best | 0.061870 | 0.018418 |
| Worst | 0.118445 | 0.065712 |
| Average | 0.097470 | 0.036136 |
| Median | 0.103960 | 0.036085 |
| Std.Dev. | 0.016750 | 0.010977 |

Table 6.23 – Inverted Generational Distance Metric for Test Problem 3

| Problem 3 | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 0.007084 | 0.007144 | 0.007224 | 0.004211 | 0.040726 |
| Worst | 0.018447 | 0.013389 | 0.012792 | 0.005510 | 0.083789 |
| Average | 0.009506 | 0.009277 | 0.008975 | **0.004727** | 0.055750 |
| Median | 0.009120 | 0.008642 | 0.008318 | 0.004731 | 0.054919 |
| Std.Dev. | 0.002157 | 0.001732 | 0.001437 | 0.000315 | 0.009659 |

Table 6.24 – Diversity Metric for Test Problem 3

| Problem 3 | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 0.639670 | 0.625549 | 0.631958 | 0.425481 | 0.482964 |
| Worst | 0.759070 | 0.830798 | 0.787958 | 0.554743 | 0.784499 |
| Average | 0.705365 | 0.729224 | 0.717679 | **0.471789** | 0.613799 |
| Median | 0.695655 | 0.724412 | 0.721937 | 0.463152 | 0.611752 |
| Std.Dev. | 0.032172 | 0.046666 | 0.036645 | 0.034208 | 0.078719 |

Table 6.25 – Diversity Reference Values for Test Problem 3

| Problem 3 | Deb et al. (2002) |
|---|---|
| | NSGA-II |
| Average | 0.411477 |
| Std.Dev. | 0.031496 |

Table 6.26 – Processing Time for Test Problem 3

| Problem 3 | Excel Add-ins (sec) | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 17.98 | 26.75 | 20.17 | 17.23 | 10.69 |
| Worst | 24.38 | 31.03 | 25.28 | 17.67 | 11.17 |
| Average | 21.66 | 29.06 | 23.06 | 17.29 | **10.89** |
| Median | 21.65 | 29.30 | 23.13 | 17.27 | 10.89 |
| Std.Dev. | 1.45 | 1.27 | 1.38 | 0.08 | 0.12 |

Table 6.27 – Dominated Ratio for Test Problem 3

| Problem 3 | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 0.46 | 0.47 | 0.44 | 0.22 | 0.89 |
| Worst | 0.67 | 0.66 | 0.68 | 0.43 | 1.00 |
| Average | 0.59 | 0.56 | 0.54 | **0.34** | 0.98 |
| Median | 0.58 | 0.55 | 0.55 | 0.34 | 1.00 |
| Std.Dev. | 0.05 | 0.05 | 0.05 | 0.06 | 0.03 |

Table 6.28 – Dominated Degree for Test Problem 3

| Problem 3 | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 0.36% | 0.34% | 0.49% | 0.16% | 4.54% |
| Worst | 1.33% | 1.27% | 1.44% | 0.98% | 9.03% |
| Average | 0.85% | 0.71% | 0.81% | **0.48%** | 6.45% |
| Median | 0.85% | 0.69% | 0.72% | 0.44% | 6.50% |
| Std.Dev. | 0.23% | 0.19% | 0.26% | 0.20% | 1.01% |

*Analysis and Comparison*

The MOPSO and NSGA-II Solvers' results were better than those reported by Coello Coello et al. (2004) and by Deb et al. (2002) for generational distance. The spacing results reported by Coello Coello et al. (2004) were very close for MOPSO and better for NSGA-II than the ones obtained by the solvers, but the authors did not report all the parameters used in their NSGA-II runs, and differences in parameters could explain the differences observed in the performance.

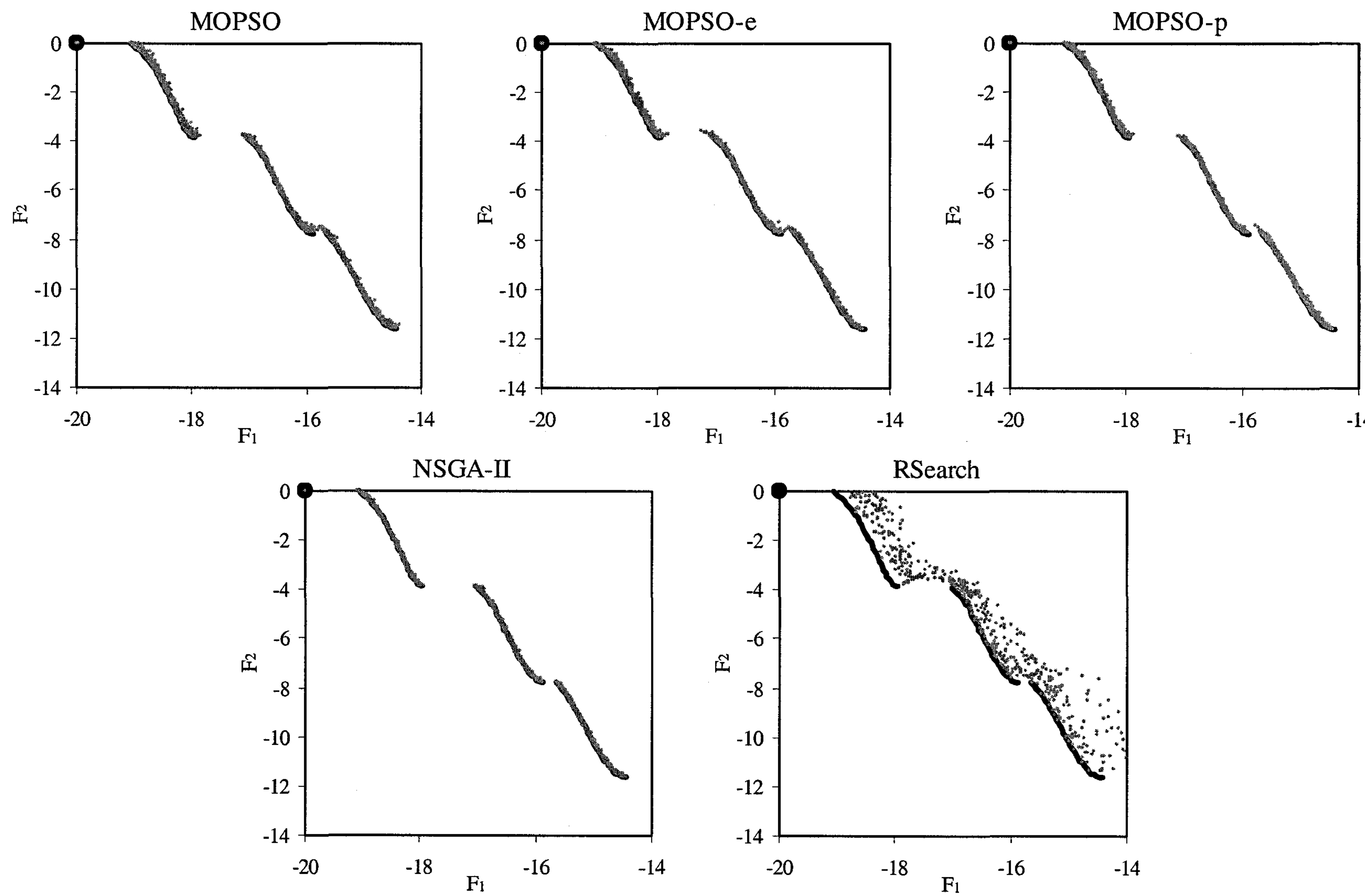



Figure 6.6 – Results for Test Problem 3

On the diversity metric the results of the NSGA-II Solver were very close to those reported by Deb et al. (2002). The dominated ratios varied from 0.34 for NSGA-II to 0.59 for MOPSO, with small dominated degrees (less than one percent) however for both solvers. Overall the NSGA-II Solver performed slightly better than the MOPSO for this problem.

### 6.1.4. Test Problem 4

This is a ten-variable problem with a convex Pareto optimal set proposed by Zitzler et al. (2000), which was used by Deb et al. (2002). This problem has about $8\text{x}10^{11}$ local Pareto optimal solutions, corresponding to $0 \leq x_1^* \leq 1$ and $x_i^* = 0.5m$, where $m$ is any integer in [-10, 10], and $i = 2, 3, \ldots, 10$. The global Pareto optimal front is obtained for $x_1^*$ in the same range [0, 1] and $x_i^* = 0$ for all $i = 2, 3, \ldots, 10$. Figure 6.7 shows the global Pareto optimal front and the best of the hundred possible local Pareto optimal fronts.

Minimize $f_1(x_i)$, (6.4)

Minimize $f_2(x_i)$

where:

$$f_1(x_i) = x_1, \qquad f_2(x_i) = g(x_i) \cdot \left[1 - \sqrt{x_1 / g(x_i)}\right], \qquad i = 1, 2, \ldots, 10$$

$$g(x_i) = 1 + 10 \cdot (n-1) + \sum_{i=2}^{n} \left[x_i^2 - 10\cos(4 \cdot \pi \cdot x_i)\right]$$

$n = 10,\ \ 0 \leq x_1 \leq 1,\ \ -5 \leq x_i \leq 5$ for $i = 2, 3, \ldots, 10$

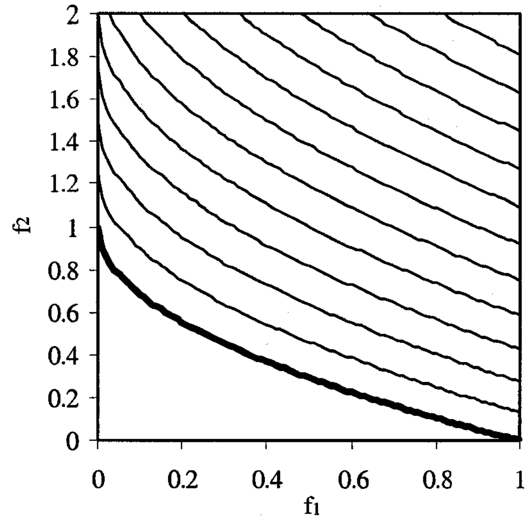



Figure 6.7 – Global and Local Pareto Optimal Fronts for Test Problem 4

This problem was solved using a population size of 100 and 250 generations. The same number of 25000 function evaluations was used Deb et al. (2002). The results are presented in Tables 6.29 to 6.37 and Figure 6.8.

Table 6.29 – Generational Distance Metric for Test Problem 4

| Problem 4 | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 0.220456 | 0.053971 | 0.051862 | 0.000857 | 22.935125 |
| Worst | 9.987351 | 1.747843 | 6.741466 | 0.430774 | 41.097248 |
| Average | 2.171299 | 0.459688 | 2.038320 | **0.118665** | 28.302693 |
| Median | 0.867433 | 0.378637 | 0.746093 | 0.091430 | 27.241680 |
| Std.Dev. | 2.686336 | 0.357437 | 2.235290 | 0.106464 | 3.797349 |

Table 6.30 – Generational Distance Reference Values for Test Problem 4

| Problem 4 | Deb et al. (2002) |
|---|---|
| | NSGA-II |
| Average | 0.513053 |
| Std.Dev. | 0.344180 |

Table 6.31 – Spacing Metric for Test Problem 4

| Problem 4 | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 0.000000 | 0.001266 | 0.000001 | 0.006868 | 3.362622 |
| Worst | 0.033750 | 0.030837 | 0.032542 | 0.014302 | 38.398369 |
| Average | 0.011402 | 0.018833 | 0.014242 | **0.009675** | 10.888541 |
| Median | 0.008016 | 0.019410 | 0.014053 | 0.009974 | 8.571790 |
| Std.Dev. | 0.010020 | 0.006809 | 0.009928 | 0.001849 | 7.669245 |

Table 6.32 – Inverted Generational Distance Metric for Test Problem 4

| Problem 4 | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 0.194534 | 0.052335 | 0.052518 | 0.000581 | 3.592583 |
| Worst | 1.523901 | 1.030993 | 1.527412 | 0.381279 | 5.487402 |
| Average | 0.704291 | 0.373528 | 0.543236 | **0.107259** | 4.683351 |
| Median | 0.646226 | 0.306698 | 0.578836 | 0.083079 | 4.686794 |
| Std.Dev. | 0.329530 | 0.235557 | 0.275093 | 0.094001 | 0.518911 |

Table 6.33 – Diversity Metric for Test Problem 4

| Problem 4 | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 0.892532 | 0.836726 | 0.900711 | 0.342079 | 0.829437 |
| Worst | 0.999998 | 0.996621 | 0.999993 | 0.868620 | 1.043785 |
| Average | 0.971743 | 0.942986 | 0.959981 | **0.706348** | 0.934098 |
| Median | 0.985259 | 0.949657 | 0.964332 | 0.734570 | 0.938743 |
| Std.Dev. | 0.028448 | 0.030008 | 0.030430 | 0.116360 | 0.057266 |

Table 6.34 – Diversity Reference Values for Test Problem 4

| Problem 4 | Deb et al. (2002) |
|---|---|
| | NSGA-II |
| Average | 0.702612 |
| Std.Dev. | 0.254260 |

Table 6.35 – Processing Time for Test Problem 4

| Problem 4 | Excel Add-ins (sec) | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 30.12 | 29.14 | 25.20 | 48.75 | 27.61 |
| Worst | 182.77 | 233.36 | 202.66 | 50.64 | 29.09 |
| Average | 100.75 | 163.44 | 102.13 | 49.52 | **27.99** |
| Median | 107.42 | 167.79 | 100.35 | 49.66 | 28.00 |
| Std.Dev. | 48.31 | 42.18 | 54.98 | 0.49 | 0.28 |

Table 6.36 – Dominated Ratio for Test Problem 4

| Problem 4 | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 1.00 | 1.00 | 1.00 | 0.27 | 1.00 |
| Worst | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| Average | 1.00 | 1.00 | 1.00 | **0.98** | 1.00 |
| Median | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| Std.Dev. | 0.00 | 0.00 | 0.00 | 0.13 | 0.00 |

Table 6.37 – Dominated Degree for Test Problem 4

| Problem 4 | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 134.66% | 40.00% | 40.02% | 0.16% | 3919.68% |
| Worst | 777.44% | 600.12% | 734.99% | 242.18% | 5671.55% |
| Average | 354.27% | 225.31% | 277.24% | **75.78%** | 4623.82% |
| Median | 310.23% | 199.89% | 268.41% | 62.52% | 4647.47% |
| Std.Dev. | 165.39% | 135.78% | 138.64% | 59.68% | 429.81% |

*Analysis and Comparison*

The NSGA-II performed much better than MOPSO in all metrics for this problem. The NSGA-II Solver results were also better than the ones reported by Deb et al. (2002).

MOPSO could not find the global Pareto optimal front in any of the thirty runs. NSGA-II found the global optimal front in only one of the thirty runs, but consistently got better local optimal fronts (closer to the global) than MOPSO.

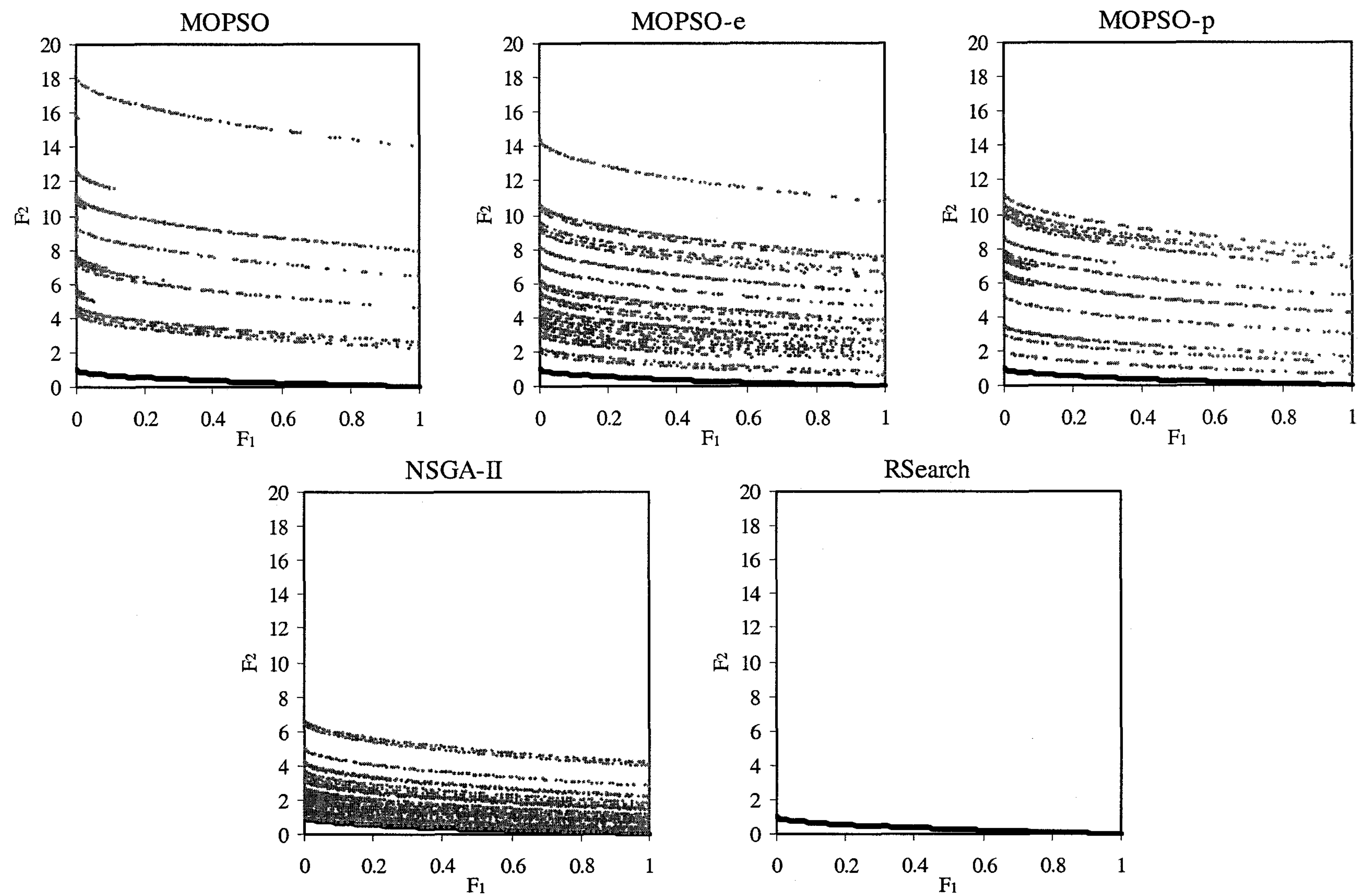



Figure 6.8 – Results for Test Problem 4

The poor performance of MOPSO as compared to NSGA-II may be explained by the way MOPSO evolves from one cycle to the next. Given the large number of local optima, during the exploratory phase of MOPSO when mutation is being applied, the algorithm often finds only one non-dominated solution in each cycle. This solution will be the global driver for all particles in the next cycle, increasing the likelihood of finding a new non-dominated solution close to the previous one. This will hinder the MOPSO exploratory capacity and its ability to produce a good spread of solutions as well.

The crossover operator performs much better for this problem. This is confirmed by the fact that the MOPSO-e, which applies a localized crossover on the extreme solutions, performed considerably better than MOPSO and MOPSO-p. Pure random search shows a very poor performance in all metrics.

### 6.1.5. Test Problem 5

This is a two-variable, two-objective problem proposed by Tanaka (1995) and used by Deb et al. (2002). The Pareto optimal front is non-convex, disconnected, and lies on the boundary of the periodic constraint imposed on the problem. Figure 6.9 shows the feasible region and the Pareto optimal front in the same plot.

Minimize $f_1(x_1, x_2)$, (6.5)

Minimize $f_2(x_1, x_2)$

where:

$$f_1(x_1, x_2) = x_1, \qquad f_2(x_1, x_2) = x_2$$

subject to:

$$C_1(x_1, x_2) \equiv x_1^2 + x_2^2 - 1 - 0.1\cos\left[16\arctan\left(\frac{x_1}{x_2}\right)\right] \geq 0$$

$$C_2(x_1, x_2) \equiv (x_1 - 0.5)^2 + (x_2 - 0.5)^2 \leq 0.5$$

$$0 \leq x_1, x_2 \leq \pi$$

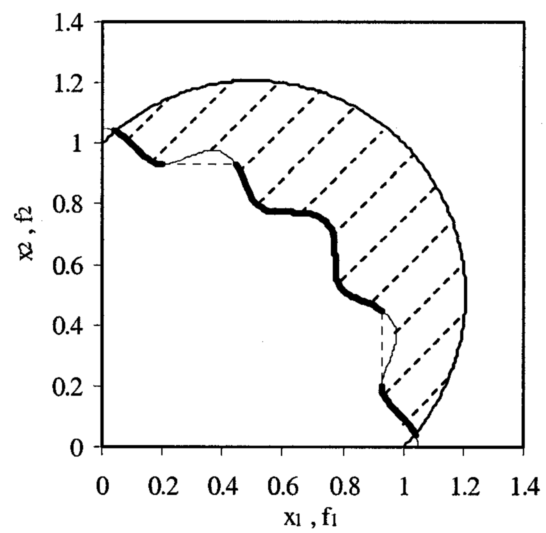



Figure 6.9 – Feasible Region and Pareto Optimal Front for Problem 5

This problem was solved using a population size of 100 and 100 generations. Deb et al. (2002) used 500 generations to test if a good spread would be maintained for a rather large number of generations. The authors highlighted, however, that less generations should already provide good results, but did not report any performance metrics for this problem. The results are presented in Tables 6.38 to 6.44 and Figure 6.10.

Table 6.38 – Generational Distance Metric for Test Problem 5

| Problem 5 | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 0.000552 | 0.000500 | 0.000417 | 0.000551 | 0.001439 |
| Worst | 0.000783 | 0.000660 | 0.000509 | 0.003597 | 0.002647 |
| Average | 0.000642 | 0.000579 | **0.000457** | 0.001015 | 0.001883 |
| Median | 0.000634 | 0.000583 | 0.000453 | 0.000732 | 0.001853 |
| Std.Dev. | 0.000049 | 0.000038 | 0.000025 | 0.000713 | 0.000314 |

Table 6.39 – Spacing Metric for Test Problem 5

| Problem 5 | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 0.005992 | 0.006155 | 0.006165 | 0.004550 | 0.009822 |
| Worst | 0.009609 | 0.009121 | 0.011045 | 0.020943 | 0.023972 |
| Average | **0.007395** | 0.007583 | 0.007737 | 0.008078 | 0.014504 |
| Median | 0.007336 | 0.007520 | 0.007483 | 0.006897 | 0.013870 |
| Std.Dev. | 0.000854 | 0.000766 | 0.001057 | 0.004206 | 0.003288 |

Table 6.40 – Inverted Generational Distance Metric for Test Problem 5

| Problem 5 | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 0.000784 | 0.000806 | 0.000777 | 0.001199 | 0.001570 |
| Worst | 0.001555 | 0.001328 | 0.001653 | 0.015901 | 0.002735 |
| Average | 0.001014 | **0.000977** | 0.001003 | 0.004888 | 0.001971 |
| Median | 0.000998 | 0.000952 | 0.000965 | 0.002670 | 0.001863 |
| Std.Dev. | 0.000158 | 0.000114 | 0.000171 | 0.004501 | 0.000311 |

Table 6.41 – Diversity Metric for Test Problem 5

| Problem 5 | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 0.768252 | 0.782421 | 0.853137 | 0.755457 | 0.643145 |
| Worst | 0.914299 | 0.917968 | 1.078403 | 1.178969 | 0.876383 |
| Average | 0.838376 | 0.848302 | 0.912273 | 0.889795 | **0.762563** |
| Median | 0.840222 | 0.845211 | 0.904882 | 0.861444 | 0.769756 |
| Std.Dev. | 0.036146 | 0.036817 | 0.047340 | 0.097876 | 0.048812 |

Table 6.42 – Processing Time for Test Problem 5

| Problem 5 | Excel Add-ins (sec) | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 6.98 | 9.77 | 7.41 | 12.59 | 10.11 |
| Worst | 8.34 | 11.06 | 8.66 | 13.33 | 10.48 |
| Average | **7.73** | 10.43 | 8.11 | 12.91 | 10.26 |
| Median | 7.74 | 10.46 | 8.11 | 12.92 | 10.25 |
| Std.Dev. | 0.31 | 0.32 | 0.28 | 0.12 | 0.11 |

Table 6.43 – Dominated Ratio for Test Problem 5

| Problem 5 | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 0.44 | 0.34 | 0.06 | 0.35 | 0.65 |
| Worst | 0.62 | 0.56 | 0.17 | 0.61 | 0.91 |
| Average | 0.53 | 0.46 | **0.10** | 0.47 | 0.82 |
| Median | 0.52 | 0.47 | 0.09 | 0.48 | 0.84 |
| Std.Dev. | 0.05 | 0.06 | 0.02 | 0.06 | 0.06 |

Table 6.44 – Dominated Degree for Test Problem 5

| Problem 5 | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 0.46% | 0.31% | 0.06% | 0.38% | 0.98% |
| Worst | 1.16% | 2.02% | 3.25% | 3.11% | 2.13% |
| Average | 0.70% | **0.69%** | 0.75% | 1.05% | 1.57% |
| Median | 0.70% | 0.56% | 0.10% | 0.79% | 1.52% |
| Std.Dev. | 0.19% | 0.36% | 0.90% | 0.76% | 0.30% |

*Analysis and Comparison*

MOPSO performed better than NSGA-II in all metrics, especially with regard to convergence. As previously mentioned, the true Pareto optimal solutions for Test Problem 5 reside on three disconnected portions, on the boundary of a periodic constraint. Evolutionary algorithms will often be more attracted to one specific portion of the Pareto front and eventually miss another portion, partly or totally. This happened with MOPSO, MOPSO-e, MOPSO-p, and particularly with NSGA-II, and it is confirmed, for example, by the fact that the iGD metrics are usually much higher than the GD metrics. This behavior is not observed with the RSearch solver since the search is completely random and the solutions are always found independently.

This problem also reveals a very interesting difference between MOPSO and NSGA-II. Although the algorithms use exactly the same approach to handle constraints, the

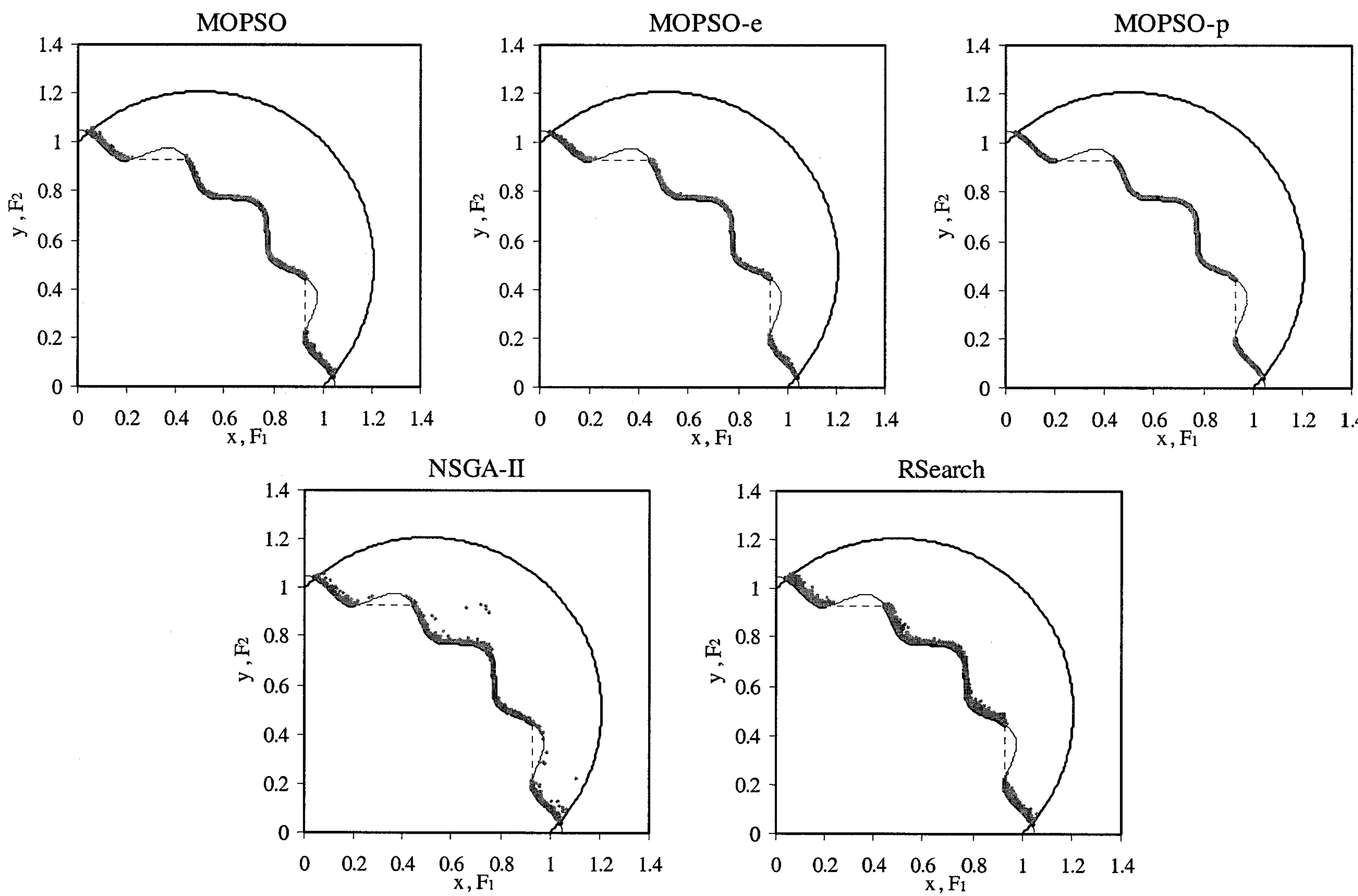



Figure 6.10 – Results for Test Problem 5

impact of those constraints on the search mechanism may be quite different. NSGA-II efficiency drops significantly when a large number of non-dominated fronts exist (the worst case is when the number of fronts equal the population size, i.e. each front with only one solution). This is exactly what happens if most of the population is infeasible with different degrees of infeasibility. In the Test Problem 5, only about five percent of the bounded space, $x_1$ and $x_2$ in $[0, \pi]$, is feasible. Because of that, Test Problem 5 is one the few for which MOPSO is actually faster than NSGA-II (despite the fact that the NSGA-II uses very efficient routines while the MOPSO implemented in this research can be significantly improved in terms of computational efficiency). For Test Problem 1, which is also a constrained problem, the NSGA-II was a little faster than MOPSO but in that case about 66% of the bounded decision space is feasible and the impact on NSGA-II efficiency is much smaller.

The dominated ratio was 0.20 for MOPSO-p and around 0.50 for MOPSO, MOPSO-e, and NSGA-II. The dominated degrees were all relatively small, with NSGA-II slightly over 1% and MOPSO around 0.7%.

### 6.1.6. Analysis of Results for the Test Functions

The results obtained by the MOPSO and NSGA-II Solvers were validated by comparison to results reported by Coello Coello et al. (2004) and Deb et al. (2002), using common metrics and parameters.

The MOPSO Solver worked a little better than NSGA-II Solver in the constrained cases, Test Problems 1 and 5, while the NSGA-II presented somewhat better results for Test Problems 2 and 3, with a much better performance for Test Problem 4 than MOPSO.

Table 6.45 presents the number of times a method presented the best average value on each performance metric.

Table 6.45 – Number of times each method showed the best average metric

| Metric | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| GD | | 1 | 1 | 3 | |
| SP | 1 | 2 | | 2 | |
| iGD | | 1 | 1 | 3 | |
| DI | | | 1 | 3 | 1 |
| PT | 1 | | | 1 | 3 |
| DR | | | 2 | 3 | |
| DD | | 2 | 1 | 2 | |

### 6.2. Selected Water Resources Management Problems

For all water resources test problems (unless specified) the MOPSO Solver was run with parameters $w = 0.4$, $c_1 = c_2 = 1$, $MutRate = 0.5$, and the NSGA-II Solver with parameters $P_{cross} = 0.9$, $P_{mut} = 1/\#$ decision variables, $\eta_c = 20$, and $\eta_m = 20$. The RSearch Solver was set to generate 30000 solutions.

#### 6.2.1. Multi-Purpose Reservoir Operational Planning

Multi-objective optimization is used to find non-dominated solutions for the operation of a single reservoir with up to four of the following objectives: (i) maximize annual firm water supply, (ii) maximize annual firm energy production, (iii) minimize flood risk, and (iv) maximize the overall reliability of the system.

The reservoir simulation model RESIMWP (Fontane 2002) performs the water balance in a monthly time step for a sequence of five years of monthly inflows. In this example, 34 years of monthly flow data were used which allows the user to select 30 different sequences of five years to be simulated. The flow data is taken from McMahon and Mein (1986, p. 347).

The following parameters must be provided: (i) Monthly water use coefficients; (ii) Monthly energy use coefficients; (iii) Parameters $a$ and $b$ of the area-volume relation – $Area(t) = a \cdot [V(t)]^b$; (iv) Monthly average evaporation depths in meters; (v) Parameters $c$, $d$, and $e$ for head calculations – $Head(t) = c \cdot [V(t)]^d + e$; and (vi) Maximum flow through turbines and minimum required head. Monthly water and energy use coefficients are used to convert annual demands to the monthly time scale. The required water releases to meet the monthly energy demands are then calculated and the water balance of the reservoir is performed as follows:

$$V(t+1) = V(t) + Inflow(t) - Evap(t) - \mathrm{Re}l(t) \tag{6.6}$$

$$Evap(t) = mthE \cdot Area(t) \tag{6.7}$$

$$\mathrm{Re}l(t) = \max\{Wrel(t), Erel(t)\} \tag{6.8}$$

where $V$ is the volume at the beginning of the month $t$, $mthE$ are monthly evaporation depths, $Wrel$ and $Erel$ are the required releases to meet water and energy monthly demands, respectively. As one can see from Eq. 6.7, the evaporated volume is calculated as a function of the water surface area at the beginning of the month only, which is ultimately a function of the volume at the beginning of the month. This simplification is acceptable as long as the volume does not vary significantly in any single month and the evaporation rates are not too high.

Three test problems were constructed using the RESIMWP model: (i) maximize firm water and firm energy; (ii) maximize firm water, firm energy, and flood control; and (iii) maximize firm water, firm energy, flood control, and overall reliability.

*Test Problem WE*

For the water energy trade-off analysis, the annual firm water and the annual firm energy are at the same time the objectives to be maximized and also the decision variables. In this case, the reservoir's active volume is fixed.

Maximize $f_1(w, e)$, (6.9)

Maximize $f_2(w, e)$

where:

$$f_1(w,e) = w, \qquad f_2(w,e) = e$$

subject to:

$$C_1 \equiv \mathrm{Re}\,l(t) \geq w \cdot wuc(t)$$

$$C_2 \equiv \mathrm{Re}\,l(t) \geq e \cdot euc(t)$$

$$C_3 \equiv V(t) \geq 0$$

$$C_4 \equiv V(61) = V_{active}$$

$$t = 1, 2, \ldots, 61; \quad 0 \leq w \leq 1000; \quad 0 \leq e \leq 35000$$

Where $w$ is the annual firm water in million cubic meters (Mcm), $e$ is the annual firm energy in megawatt-hour (Mwh), $wuc(t)$ is the water use coefficient for month $t$, $euc(t)$ is the energy use coefficient for month $t$, and $V_{active}$ is the active volume of the reservoir in Mcm. The underlying assumption to constraint $C_4$ is that the reservoir is full at the end of the five-year simulation (beginning volume at month 61). It is also assumed that the reservoir is full at the beginning $V(1) = V_{active}$. The Pareto optimal front for this problem, obtained using the ε-NLP Solver described in the next chapter, is shown in Figure 6.11.

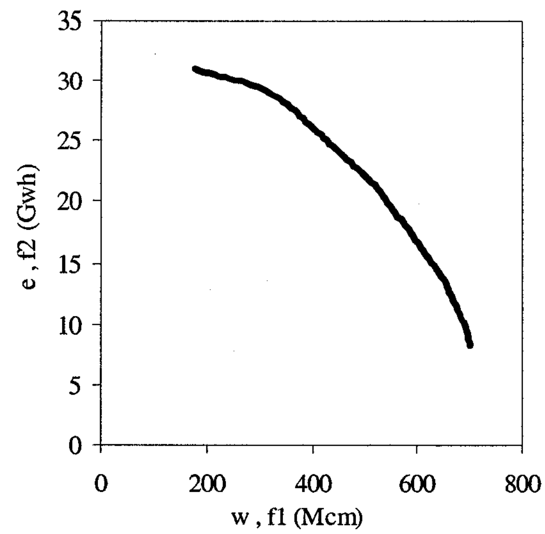



Figure 6.11 – Pareto Optimal Front for Test Problem WE

This problem was solved using a population size of 100 and 100 generations. The results are presented in Tables 6.46 to 6.52 and Figure 6.12. The metrics were calculated using the ε-NLP Solver solution as the true Pareto optimal set.

Table 6.46 – Generational Distance Metric for Test Problem WE

| Problem WE | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 6.918785 | 6.376023 | 6.182394 | 6.559626 | 6.937461 |
| Worst | 90.716125 | 17.155519 | 7.757367 | 8.506394 | 98.750610 |
| Average | 18.423611 | 8.109958 | **7.194375** | 7.410331 | 40.229682 |
| Median | 12.340032 | 7.342797 | 7.227124 | 7.263885 | 32.163016 |
| Std.Dev. | 17.435010 | 2.328121 | 0.413530 | 0.507003 | 28.727062 |

Table 6.47 – Spacing Metric for Test Problem WE

| Problem WE | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 100.063011 | 95.679657 | 99.257996 | 88.152039 | 86.807510 |
| Worst | 394.380127 | 233.567215 | 165.018066 | 149.419830 | 523.618347 |
| Average | 157.136276 | 130.267808 | 121.859362 | **118.422367** | 229.367990 |
| Median | 140.816742 | 122.200775 | 115.727161 | 116.349064 | 201.382607 |
| Std.Dev. | 64.466856 | 33.834105 | 17.094626 | 14.887482 | 127.107528 |

Table 6.48 – Inverted Generational Distance Metric for Test Problem WE

| Problem WE | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 11.224649 | 10.965480 | 11.379469 | 10.325263 | 10.040552 |
| Worst | 23.975431 | 21.157345 | 20.598803 | 20.705256 | 16.580736 |
| Average | 13.910921 | 14.423551 | 13.179414 | 14.136986 | **11.658912** |
| Median | 13.030120 | 13.611643 | 12.628768 | 13.137642 | 11.461673 |
| Std.Dev. | 2.798601 | 2.353840 | 1.791330 | 2.888022 | 1.438950 |

Table 6.49 – Diversity Metric for Test Problem WE

| Problem WE | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 0.596296 | 0.576088 | 0.576892 | 0.446448 | 0.512727 |
| Worst | 0.770524 | 0.741649 | 0.717984 | 0.605663 | 0.770321 |
| Average | 0.678873 | 0.669279 | 0.652298 | **0.531487** | 0.642741 |
| Median | 0.675885 | 0.672927 | 0.651328 | 0.538321 | 0.645760 |
| Std.Dev. | 0.043534 | 0.038936 | 0.034809 | 0.040722 | 0.076478 |

Table 6.50 – Processing Time for Test Problem WE

| Problem WE | Excel Add-ins (sec) | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 29.38 | 47.59 | 33.44 | 30.44 | 69.48 |
| Worst | 31.73 | 51.09 | 35.61 | 31.67 | 72.53 |
| Average | **30.15** | 48.90 | 34.30 | 30.55 | 70.19 |
| Median | 30.00 | 48.87 | 34.28 | 30.49 | 69.86 |
| Std.Dev. | 0.47 | 0.77 | 0.52 | 0.23 | 0.83 |

Table 6.51 – Dominated Ratio for Test Problem WE

| Problem WE | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 0.45 | 0.45 | 0.02 | 0.13 | 0.56 |
| Worst | 0.68 | 0.63 | 0.09 | 0.34 | 0.72 |
| Average | 0.56 | 0.52 | **0.05** | 0.24 | 0.64 |
| Median | 0.56 | 0.51 | 0.04 | 0.24 | 0.65 |
| Std.Dev. | 0.07 | 0.05 | 0.02 | 0.05 | 0.04 |

Table 6.52 – Dominated Degree for Test Problem WE

| Problem WE | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 0.47% | 0.41% | 0.01% | 0.19% | 0.67% |
| Worst | 1.55% | 0.85% | 0.07% | 0.41% | 1.62% |
| Average | 0.74% | 0.53% | **0.04%** | 0.30% | 1.07% |
| Median | 0.67% | 0.52% | 0.04% | 0.28% | 1.04% |
| Std.Dev. | 0.24% | 0.09% | 0.01% | 0.06% | 0.25% |

Analysis and Comparison

The MOPSO-p presented better results in terms of convergence (GD, iGD) with diversity metrics slightly worse than NSGA-II. Non-optimal solutions on the extremities of the Pareto curve cause the GD metric for the basic MOPSO to be considerably worse than the NSGA-II. All evolutionary solvers presented similar iGD metrics while the RSearch Solver provided the best value for this metric. The iGD values are considerably larger than the GD values for MOPSO-e, MOPSO-p, and NSGA-II, which indicates that these algorithms are eventually failing to explore the whole extension of the Pareto curve.

The dominated ratio varied from 0.05 for MOPSO-p to 0.56 for MOPSO, with small dominated degrees (< 1%), indicating that all methods provided solutions that are in average very close to optimal. Overall, the results obtained by pure random search (RSearch Solver) were similar to those of the other methods, with exception of the GD which was much worse.

Best Compromise Solutions

The best compromise solutions for three different sets of weights are presented in Table 6.53 and Figure 6.13.

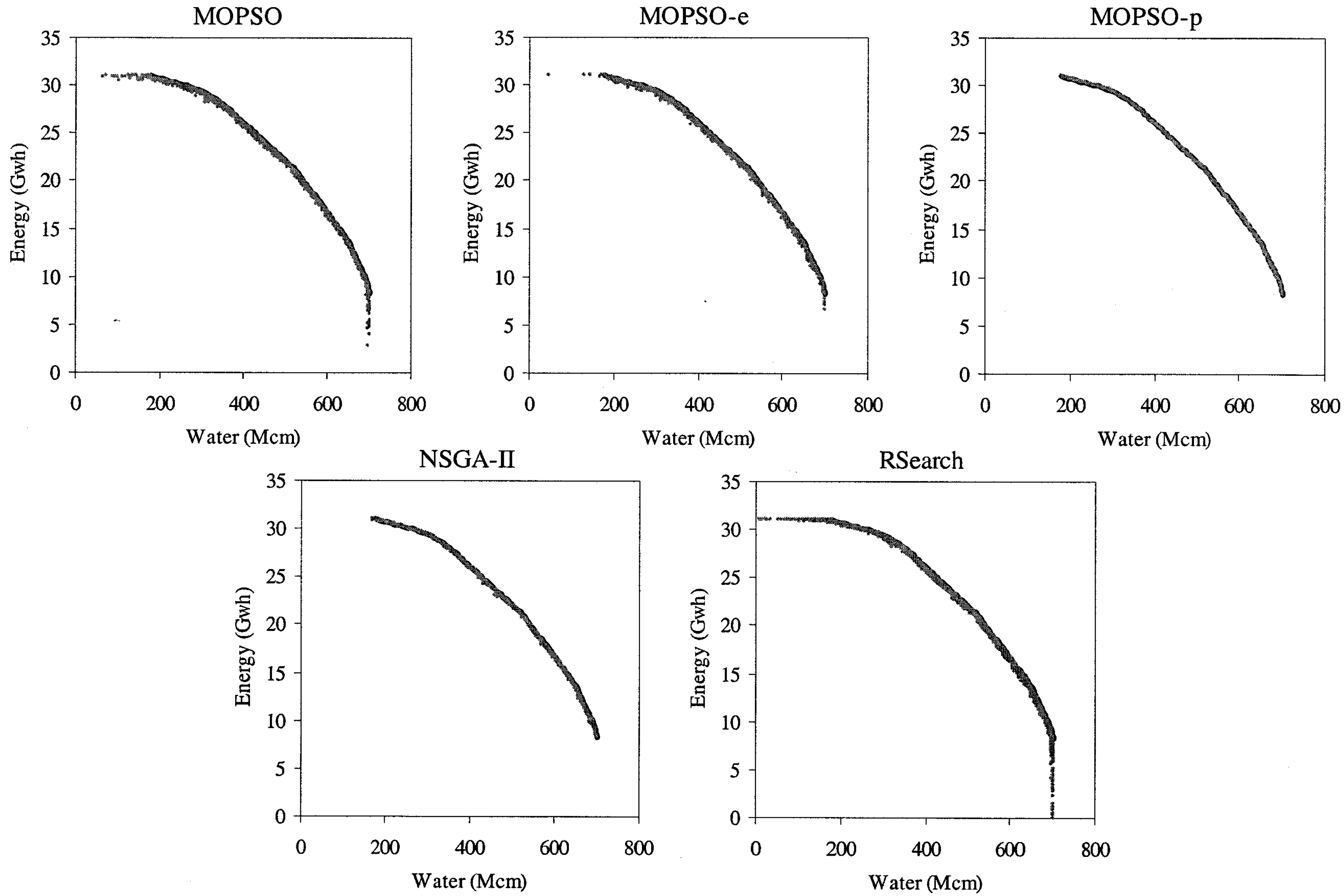



Figure 6.12 – Results for Test Problem WE

Table 6.53 – Best Compromise Solutions for Test Problem WE

| Case | W [Mcm] | E [Mwh] |
|---|---|---|
| (a) Weight [1,1] | 507.6 | 21800.9 |
| (b) Weight [1,0] | 700.0 | 8264.5 |
| (c) Weight [0,1] | 177.3 | 30978.2 |

* Weights: [$W_W$,$W_E$]

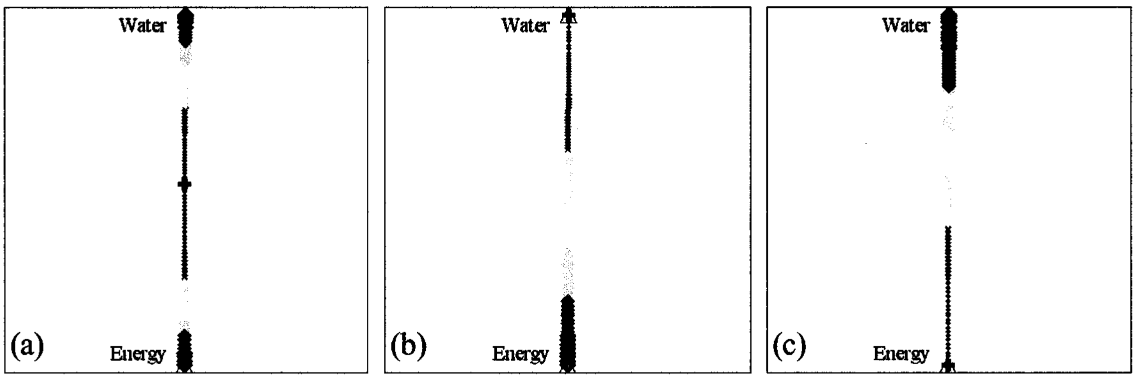



Figure 6.13 – Best Compromise Solutions for Test Problem WE

*Test Problem WEF*

This problem is obtained by adding a flood control objective to the previous problem. The flood control objective is introduced by maximizing the flood space, which is now also an additional decision variable. The active volume is computed by subtracting the flood space from the reservoir capacity (800 Mcm). For simplicity, the flood space is applied to all months indistinctly, which is not the way reservoirs are usually operated for flood control. Increasing flood spaces correspond to reduced active volumes and more restricted water and energy productions.

$$\text{Maximize } f_1(w, e, fs), \tag{6.10}$$

$$\text{Maximize } f_2(w, e, fs),$$

$$\text{Maximize } f_3(w, e, fs),$$

where:

$$f_1(w,e,fs) = w, \qquad f_2(w,e,fs) = e, \qquad f_3(w,e,fs) = fs$$

subject to:

All constraints of Test Problem WE, and

$0 \leq fs \leq 400$

where *fs* is the flood space in Mcm.

The Pareto optimal surface obtained by the ε-NLP Solver is shown in Figure 6.14. Figure 6.15 shows the same Pareto optimal solutions as displayed using the ICC method. The alternatives marked with a cross are the best compromise solutions, with equal weights for all objectives. Larger markers correspond to worse alternatives. The color convention goes from light green (best) to dark red (worst), with a total of seven categories (the same color convention is used in Figures 6.14 and 6.15).

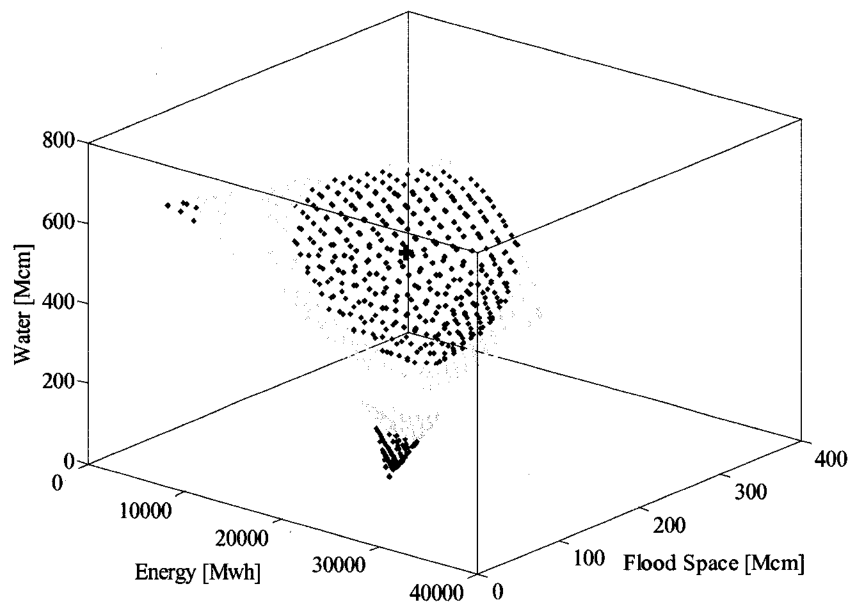



Figure 6.14 – Pareto Optimal Surface for Test Problem WEF

From an optimization viewpoint, Test Problems WE and WEF are not particularly complex. The objectives are linearly related to the decision variables, although the constraints are highly nonlinear due to energy and evaporation computation.

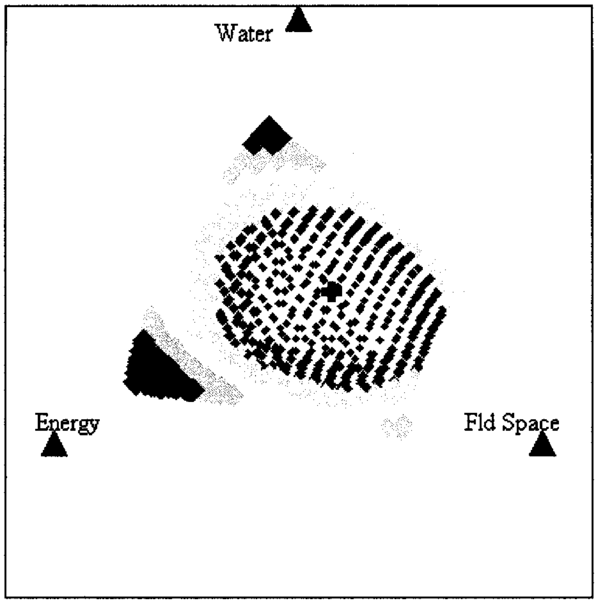



Figure 6.15 – ICC Graph of Pareto Optimal Set for Test Problem WEF

This problem was solved using a population size of 150 and 150 generations, with $c_1 = c_2 = 1.4$ for MOPSO. The results are shown in Tables 6.54 to 6.59 and Figure 6.16. The metrics were obtained using the ε-NLP Solver solution as the true Pareto optimal set.

Table 6.54 – Generational Distance Metric for Test Problem WEF

| Problem WEF | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 41.948223 | 6.669309 | 6.104114 | 5.792999 | 42.490265 |
| Worst | 77.782440 | 67.601547 | 7.085696 | 8.568276 | 66.024544 |
| Average | 63.595616 | 20.464939 | **6.574283** | 6.669624 | 53.507358 |
| Median | 65.437790 | 8.629755 | 6.547800 | 6.656199 | 53.177269 |
| Std.Dev. | 9.636843 | 19.425531 | 0.246186 | 0.530888 | 6.284155 |

Table 6.55 – Spacing Metric for Test Problem WEF

| Problem WEF | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 137.645508 | 111.722092 | 114.538551 | 123.494659 | 132.673370 |
| Worst | 180.758514 | 480.569611 | 182.956253 | 179.760025 | 172.677887 |
| Average | 157.768608 | 194.058207 | **138.526761** | 144.332518 | 151.239996 |
| Median | 158.800812 | 160.811714 | 136.655533 | 142.504089 | 152.496719 |
| Std.Dev. | 11.202391 | 99.458431 | 14.756336 | 12.613192 | 10.913420 |

Table 6.56 – Inverted Generational Distance Metric for Test Problem WEF

| Problem WEF | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 6.286863 | 5.875835 | 5.909912 | 5.539914 | 6.044873 |
| Worst | 8.452724 | 19.786551 | 7.495313 | 7.571140 | 8.357879 |
| Average | 7.072915 | 7.686200 | 6.673925 | **6.302878** | 7.263186 |
| Median | 6.998417 | 6.822933 | 6.610275 | 6.277871 | 7.238054 |
| Std.Dev. | 0.515336 | 2.600944 | 0.416269 | 0.489064 | 0.547650 |

Table 6.57 – Processing Time for Test Problem WEF

| Problem WEF | Excel Add-ins (sec) | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 241.94 | 404.97 | 245.04 | 115.90 | 121.75 |
| Worst | 250.77 | 424.94 | 261.48 | 118.18 | 127.98 |
| Average | 246.39 | 413.89 | 252.38 | **116.90** | 124.01 |
| Median | 245.93 | 413.53 | 252.53 | 116.86 | 123.66 |
| Std.Dev. | 2.48 | 5.04 | 4.40 | 0.55 | 1.25 |

Table 6.58 – Dominated Ratio for Test Problem WEF

| Problem WEF | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 0.61 | 0.53 | 0.00 | 0.14 | 0.44 |
| Worst | 0.80 | 0.74 | 0.01 | 0.26 | 0.61 |
| Average | 0.73 | 0.63 | **0.00** | 0.21 | 0.53 |
| Median | 0.73 | 0.62 | 0.00 | 0.22 | 0.54 |
| Std.Dev. | 0.04 | 0.05 | 0.00 | 0.03 | 0.04 |

Table 6.59 – Dominated Degree for Test Problem WEF

| Problem WEF | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 3.44% | 2.53% | 0.00% | 1.58% | 3.31% |
| Worst | 5.18% | 3.55% | 0.04% | 3.26% | 4.74% |
| Average | 4.15% | 2.90% | **0.00%** | 2.30% | 4.00% |
| Median | 4.13% | 2.84% | 0.00% | 2.25% | 4.01% |
| Std.Dev. | 0.47% | 0.23% | 0.01% | 0.40% | 0.34% |

Analysis and Comparison

The MOPSO-p is again the one that performs the best in most metrics. The NSGA-II provided similar results for GD and spacing metrics and it was slightly better in terms of iGD. The basic MOPSO had the worst values in most metrics, being outperformed even by the pure random search.

This poor performance may be explained by the same issue observed in Test Problem WE. The MOPSO algorithm is able to cover relatively well the whole extension of the true Pareto surface (small iGD) but keeps highly non-optimal solutions on the boundaries.

The issue arises when the algorithm finds a solution which is very close to the best feasible value for one objective but sub-optimal for other(s). In this case, the range of feasible movements becomes significantly restricted and Pareto improvements are very difficult to achieve because most changes to such solutions in the decision space will likely make them infeasible. Given the formulation of this test problem, where the decision and objective spaces are actually the same, a Pareto improvement would require a feasible movement in one dimension while keeping the others mostly unchanged. The fine tuning of these movements becomes extremely difficult.

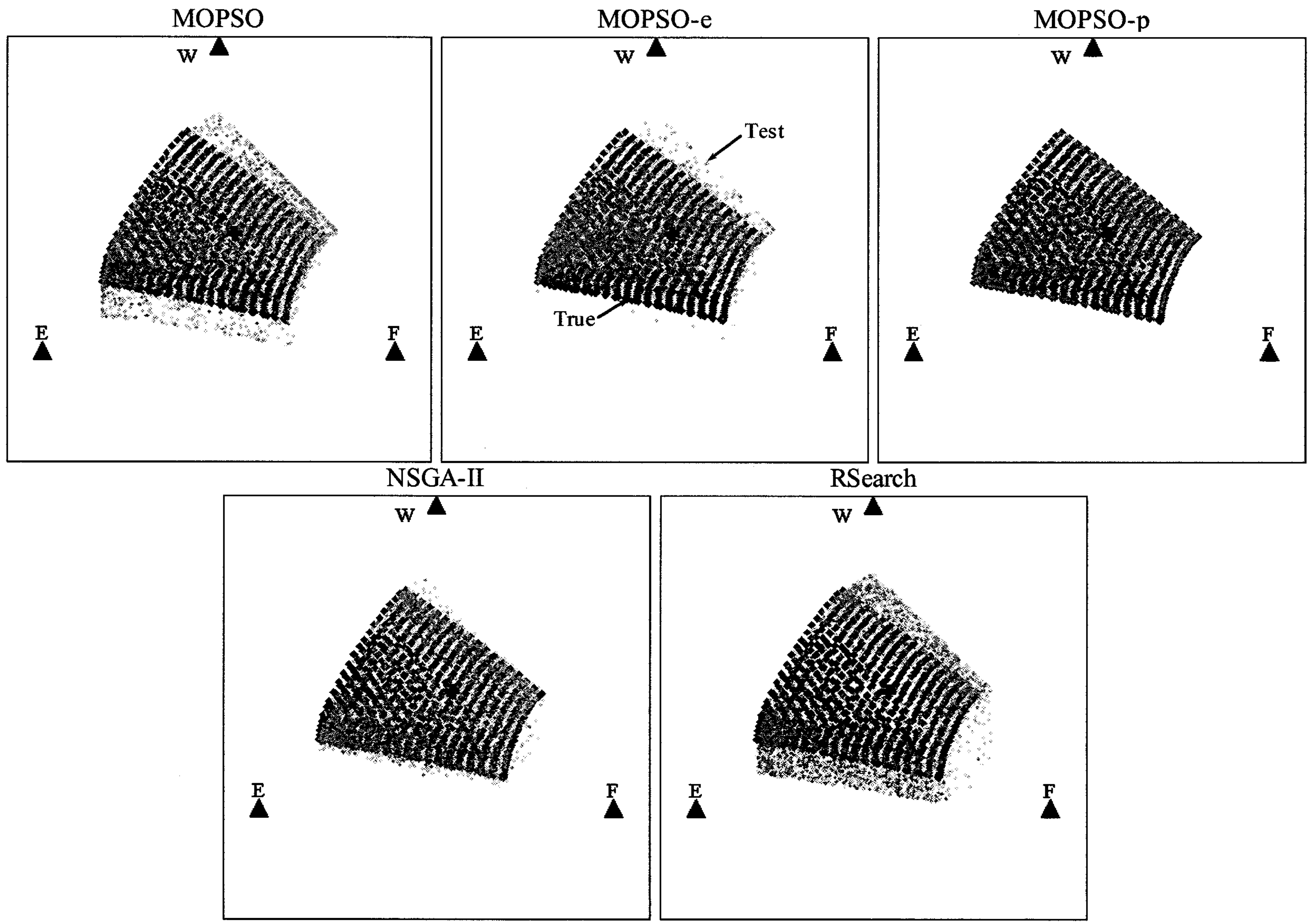



Figure 6.16 – Results for Test Problem WEF

MOPSO is especially susceptible to this problem because of the way the particles' positions in the decision space are updated in each cycle. As discussed in section 4.1, in each cycle, the MOPSO algorithm randomly updates all dimensions of each particle using the velocity equation. This approach enhances the exploratory capabilities of the MOPSO search mechanism but may compromise convergence for certain problems.

The NSGA-II crossover operator is less susceptible to the problem described above. New child solutions are obtained by recombination of characteristics of two randomly selected parent solutions. The crossover is performed on each dimension individually (each decision variable) but not all dimensions are crossed to form child solutions. In fact, for each decision variable the probability of crossover is fifty percent, i.e. each dimension has fifty percent chance of remaining unchanged.

Overall, as occurred with Test Problem WE, the performance of the pure random search is very similar to the evolutionary algorithms, even outperforming the later in some of the performance metrics.

The dominated ratio ranged from virtually zero for MOPSO-p to 0.73 for MOPSO, while the dominated degrees were below five percent for all solvers.

Best Compromise Solutions

The best compromise solutions for four different sets of weights are presented in Table 6.60 and Figure 6.17.

Table 6.60 – Best Compromise Solutions for Test Problem WEF

| Case | W [Mcm] | E [Mwh] | F [Mcm] |
|---|---|---|---|
| (a) Weight [1,1,1]* | 431.9 | 13608.9 | 231.6 |
| (b) Weight [3,1,1] | 540.2 | 10500.7 | 188.2 |
| (c) Weight [1,3,1] | 329.7 | 21625.5 | 140.2 |
| (d) Weight [1,1,3] | 400.6 | 9600.6 | 352.9 |

* Weights: [$W_W$,$W_E$,$W_F$]

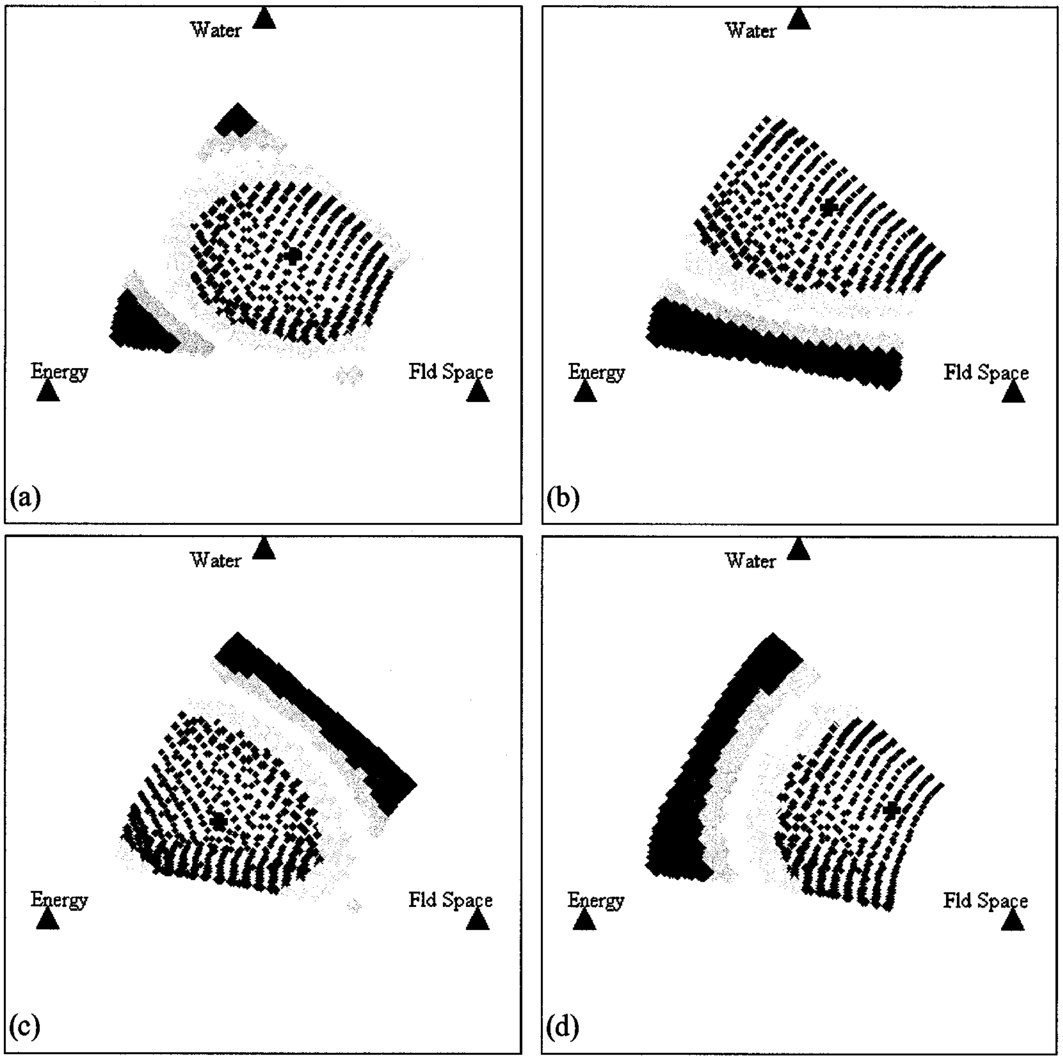



Figure 6.17 – Best Compromise Solutions for Test Problem WEF

*Test Problem WEFR*

This problem is constructed by adding a reliability objective to the previous problem. Each solution (a tri-dimensional vector – annual firm water, annual firm energy, and flood space) is evaluated for the thirty different five-year inflow sequences. The number of sequences with failures (water shortage, energy shortage, or an ending volume less than the full active volume) is counted and a reliability index is calculated by Eq. 6.11 to follow. This computation is done through a user-defined macro using the scripting tool included in the MOPSO, NSGA-II, and RSearch Solvers. No dimension is added in the search space, which is still defined by firm water, firm energy, and flood space decision variables. The Pareto front is defined in a four-dimensional objective space, however.

$$f_4(w,e,fs) = 1 - \frac{Nfail(w,e,fs)}{30} \tag{6.11}$$

The optimization problem is the same as the Test Problem WEF adding the $f_4(w, e, fs)$ objective, defined above, to be maximized.

This problem is notably more difficult to optimize than the previous two. Not only because of the additional objective but mostly due to the way this objective was defined. $Nfail(w, e, fs)$ in Eq. 6.11 is an integer number in [0, 30], which makes the $f_4$ objective assume discrete values, introducing a discontinuity in the Pareto hypersurface.

Indeed, the process to obtain a true Pareto optimal set was much more complicated than the previous ones. The ε-NLP Solver had to be executed many times with many different initial solutions in order to get a set with a reasonable number of Pareto optimal solutions. The spread of solutions was not adequate, however. The ε-NLP set was merged with all solutions obtained from 30-run sequences of the MOPSO, NSGA-II, and RSearch Solvers, with a total of about fifteen thousand potential solutions. Dominated solutions were excluded from the merged set and the Excel Solver was used again to individually check all solutions so that non-optimal solutions could be removed. One cannot guarantee that the whole extension of the possible trade-offs has been found, however. The ICC graph of the Pareto optimal set thus obtained is shown in Figure 6.18.

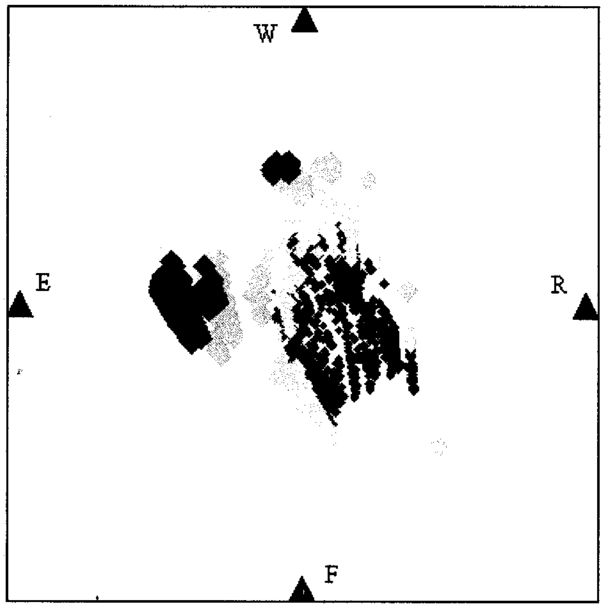



Figure 6.18 – Pareto Optimal Set for Test Problem WEFR

This problem was solved using a population size of 150 and 150 generations, with $c_1 = c_2 = 1.4$ for MOPSO. The results are shown in Tables 6.61 to 6.68 and Figure 6.19.

Given the complexity and uncertainties on the derivation of the true Pareto optimal set for this problem using nonlinear optimization, pair-wise comparisons of the methods on the dominated ratio and dominated degree metrics are presented in addition to the other metrics, which are based on the obtained "true" Pareto optimal set.

Table 6.61 – Generational Distance Metric for Test Problem WEFR

| Problem WEFR | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 12.829226 | 7.154554 | 4.331051 | 7.460876 | 20.451916 |
| Worst | 17.764570 | 19.054985 | 6.495975 | 15.268907 | 25.335693 |
| Average | 15.458379 | 9.645115 | **5.099927** | 10.665476 | 22.869765 |
| Median | 15.325740 | 8.899469 | 5.098110 | 10.491458 | 22.994317 |
| Std.Dev. | 1.385035 | 2.480729 | 0.430772 | 2.134289 | 1.286416 |

Table 6.62 – Spacing Metric for Test Problem WEFR

| Problem WEFR | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 144.152054 | 114.928116 | 110.739090 | 125.298065 | 156.376648 |
| Worst | 188.331375 | 367.274292 | 259.401703 | 208.595901 | 203.006058 |
| Average | 160.047502 | **155.936805** | 164.232261 | 168.460003 | 176.572985 |
| Median | 157.551262 | 139.773041 | 144.987022 | 168.582825 | 172.869682 |
| Std.Dev. | 11.354061 | 55.936937 | 47.820488 | 20.294896 | 13.745695 |

Table 6.63 – Inverted Generational Distance Metric for Test Problem WEFR

| Problem WEFR | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 2.724133 | 2.977720 | 2.468497 | 2.404234 | 2.939846 |
| Worst | 3.605318 | 5.524015 | 3.426756 | 2.877154 | 3.746339 |
| Average | 3.094565 | 3.541727 | 2.878444 | **2.634215** | 3.303247 |
| Median | 3.006235 | 3.386609 | 2.791884 | 2.635825 | 3.307081 |
| Std.Dev. | 0.231432 | 0.568025 | 0.250285 | 0.122282 | 0.226130 |

Table 6.64 – Processing Time for Test Problem WEFR

| Problem WEFR | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 1254.00 | 4594.01 | 1524.82 | 828.00 | 1119.29 |
| Worst | 1470.84 | 7010.59 | 1795.70 | 856.83 | 1333.16 |
| Average | 1333.85 | 5102.84 | 1714.33 | **838.16** | 1193.79 |
| Median | 1306.64 | 4731.59 | 1731.33 | 842.09 | 1144.77 |
| Std.Dev. | 70.58 | 711.80 | 65.83 | 7.70 | 83.82 |

Table 6.65 – Dominated Ratio for Test Problem WEFR

| Problem WEFR | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 0.59 | 0.35 | 0.07 | 0.43 | 0.33 |
| Worst | 0.77 | 0.61 | 0.15 | 0.61 | 0.51 |
| Average | 0.68 | 0.50 | **0.12** | 0.53 | 0.42 |
| Median | 0.68 | 0.50 | 0.11 | 0.53 | 0.42 |
| Std.Dev. | 0.04 | 0.06 | 0.02 | 0.04 | 0.04 |

Table 6.66 – Dominated Degree for Test Problem WEFR

| Problem WEFR | Excel Add-ins | | | | |
|---|---|---|---|---|---|
| | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
| Best | 4.55% | 3.76% | 0.01% | 3.13% | 3.16% |
| Worst | 6.56% | 5.66% | 0.55% | 5.23% | 4.48% |
| Average | 5.54% | 4.53% | **0.07%** | 4.00% | 3.86% |
| Median | 5.54% | 4.45% | 0.01% | 4.09% | 3.82% |
| Std.Dev. | 0.46% | 0.48% | 0.14% | 0.59% | 0.33% |

Table 6.67 – Pair-wise Comparison on Average DR for Problem WEFR

| P. WEFR* | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
|---|---|---|---|---|---|
| MOPSO | - | 0.1660 | 0.3931 | 0.1627 | 0.0629 |
| MOPSO-e | 0.0798 | - | 0.2952 | 0.0952 | 0.0396 |
| MOPSO-p | 0.0000 | 0.0002 | - | 0.0018 | 0.0000 |
| NSGA-II | 0.0505 | 0.0590 | 0.1705 | - | 0.0234 |
| RSearch | 0.0216 | 0.0231 | 0.1160 | 0.0567 | - |

* The element in the first row third column, 0.3931, means that about 39% of the solutions in the MOPSO test sets are dominated by solutions in the MOPSO-p test sets.

Table 6.68 – Pair-wise Comparison on Average DD for Problem WEFR

| P. WEFR* | MOPSO | MOPSO-e | MOPSO-p | NSGA-II | RSearch |
|---|---|---|---|---|---|
| MOPSO | - | 4.30% | 6.07% | 5.28% | 5.35% |
| MOPSO-e | 3.96% | - | 4.78% | 4.47% | 4.96% |
| MOPSO-p | 0.00% | 0.12% | - | 0.06% | 0.00% |
| NSGA-II | 4.59% | 5.12% | 5.32% | - | 4.11% |
| RSearch | 2.62% | 2.63% | 4.10% | 3.12% | - |

* The element in the first row third column, 6.07%, means that for those solutions in the MOPSO test sets that are dominated by solutions in the MOPSO-p test sets, the average degree of domination is about 6% of the ranges of the objective functions.

Analysis and Comparison

Overall, the MOPSO-p outperformed all other solvers, with much better GD, an iGD slightly worse than NSGA-II, and better values for dominated ratio and dominated degree. On the pair-wise comparisons, the MOPSO-p does considerably better than all other solvers in both dominated ratio and dominated degree.

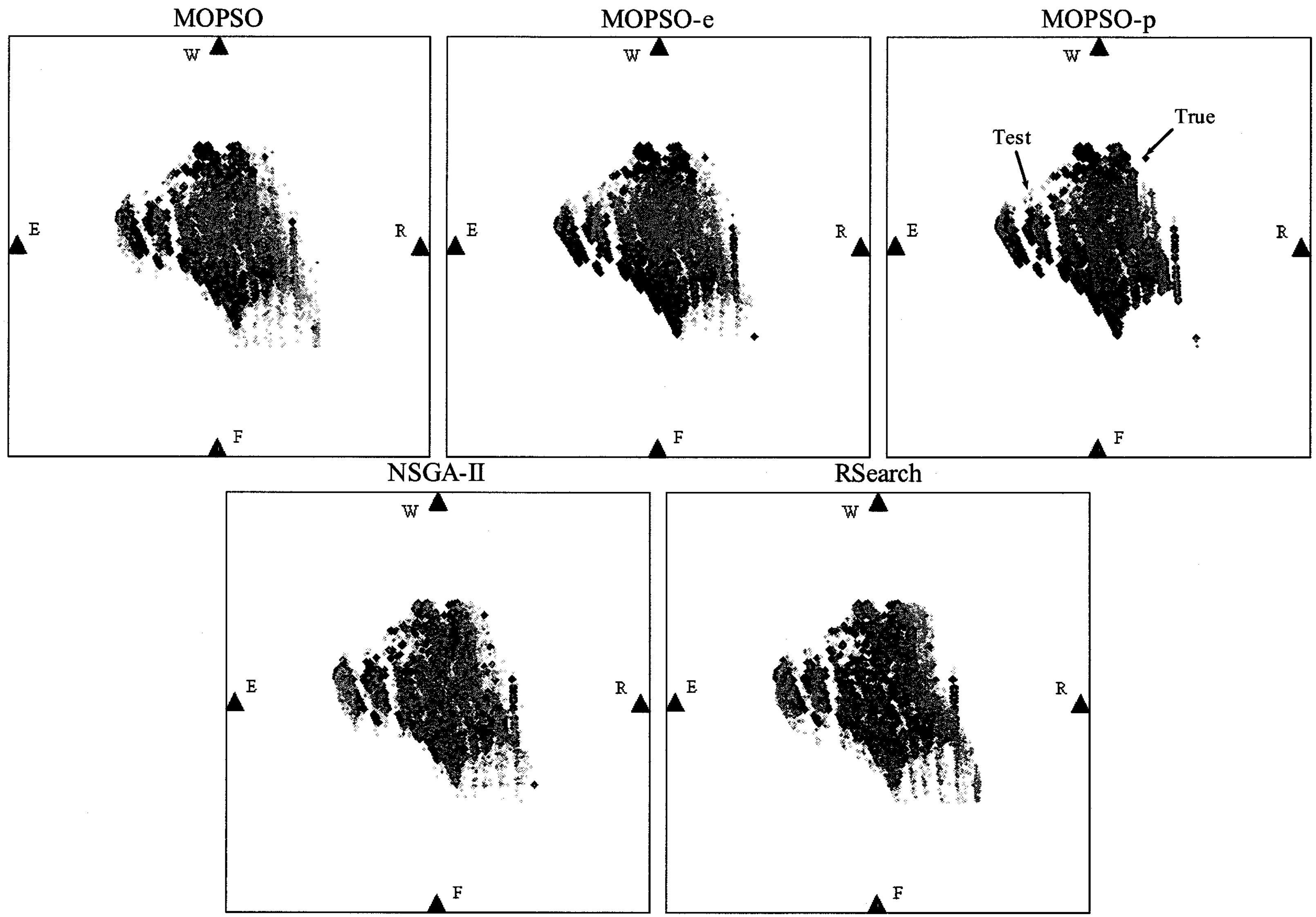



Figure 6.19 – Results for Test Problem WEFR

This test problem shows the same kind of behavior observed in the last two problems which also use the RESIMWP model: the pure random search works relatively well, even outperforming the evolutionary algorithms in some aspects, with the exception of the MOPSO-p. This issue is discussed on section 6.2.4.

The use of a localized crossover on the extreme solutions (MOPSO-e) was able to improve significantly the GD, by reducing the dominated degrees and by eliminating some of the highly non-optimal solutions kept by the MOPSO algorithm at the boundaries of the Pareto hypersurface. A worse coverage of the true Pareto set is observed however, which is reflected in the larger value of the iGD metric and graphically in Figure 6.19 as well.

Best Compromise Solutions

The best compromise solutions for four different sets of weights are presented in Table 6.69 and Figure 6.20.

Table 6.69 – Best Compromise Solutions for Test Problem WEFR

| Case | W [Mcm] | E [Mwh] | F [Mcm] | R |
|---|---|---|---|---|
| (a) Weight [1,1,1,1] | 396.8 | 10146.8 | 341.5 | 0.90 |
| (b) Weight [1,4,1,1] | 341.7 | 21001.4 | 136.6 | 0.80 |
| (c) Weight [3,1,0,0] | 650.1 | 13608.9 | 0.0 | 0.67 |
| (d) Weight [0,0,1,1] | 122.8 | 1586.3 | 400.0 | 1.00 |

* Weights: [$W_W$,$W_E$,$W_F$, $W_R$]

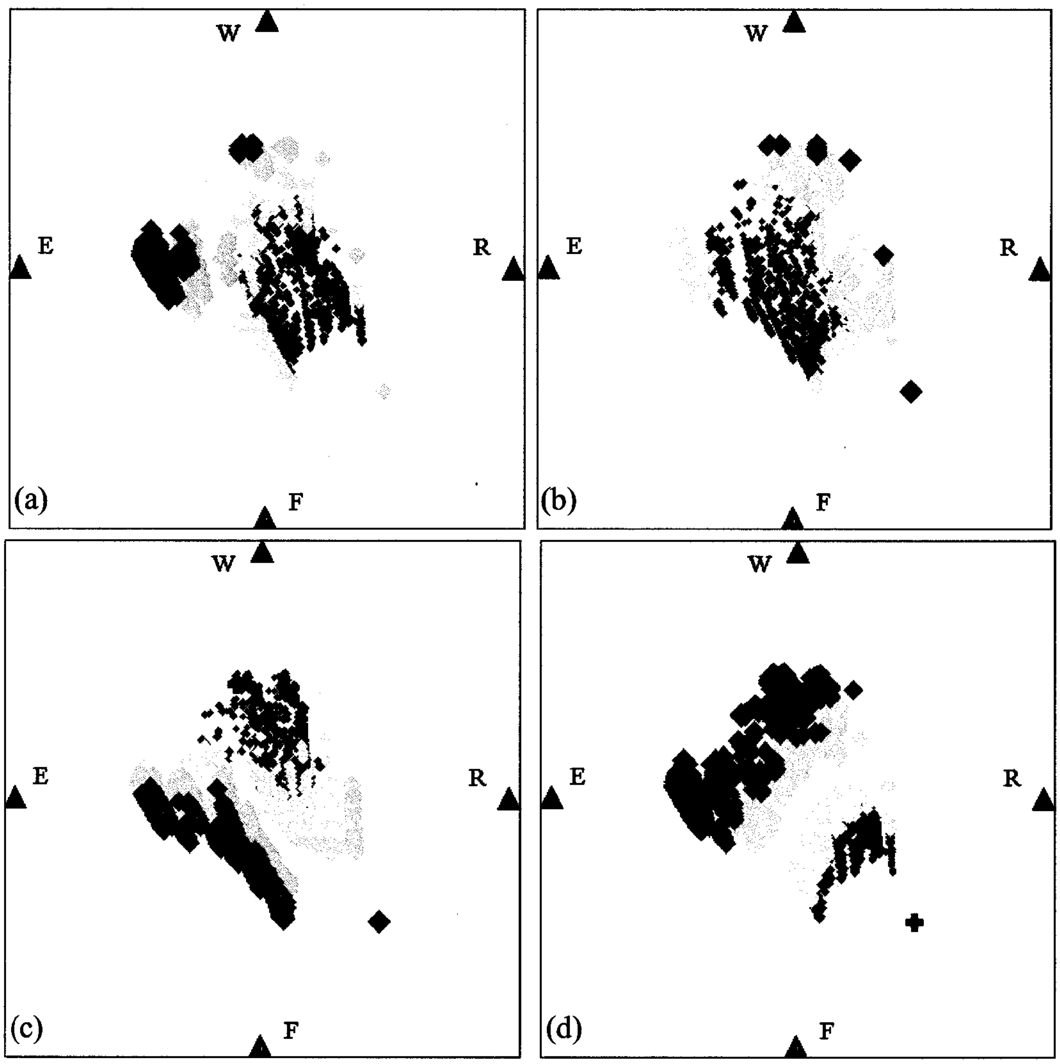



Figure 6.20 – Best Compromise Solutions for Test Problem WEFR

6.2.2. Selective Withdrawal from a Thermally Stratified Reservoir

Multi-objective optimization is used to find Pareto optimal solutions for the operation of a selective withdrawal structure. The algorithms are applied to determine flows from the various ports that minimize the square deviations from outflow water quality targets of: (i) temperature (T), (ii) dissolved oxygen (DO), and (iii) total dissolved solids (TDS).

The selective withdrawal model (SELECT) is described in detail in Bohan and Grace (1973), and Fontane and Schneider (1994). The SELECT model evaluates the release temperature, DO content and TDS for any combination of flows from different ports. The user has to provide, as main inputs, the levels where the ports are located, and the water quality profiles in the reservoir for the three mentioned variables. Using the temperature profile, the model computes the density profile which is used to determine the zone of withdrawal of each port using Eq. 6.12.

$$V_o = \frac{Z^2}{A_o}\sqrt{\left(\frac{\Delta\rho'}{\rho_o}\right)\cdot g \cdot Z} \tag{6.12}$$

Where $V_o$ is the average velocity through the port, $Z$ is the vertical distance from the elevation of the port center line to the lower or upper limit of the zone of withdrawal, $A_o$ is the area of the port opening, $\Delta\rho'$ is the density difference between the port center line and the lower or upper limit of the zone of withdrawal, $\rho_o$ is the density at the elevation of the port center line, and $g$ is acceleration due to gravity.

Once the zone of withdrawal is defined, the velocity profile within this zone is determined as a function of the density profile and the average velocity. Details of these calculations can be found in Bohan and Grace (1973).

The water quality characteristics of the total outflow are then computed as a flow-weighted average throughout the depth of the reservoir using the velocity profile and the quality profiles.

Fontane and Schneider (1994) used a linear goal programming approach to find the flows that would meet release quality targets as closely as possible. This application, however, produced only one solution for each set of weights introduced by the DM to penalize deviations from the quality targets. The SELECT model is first used to find quality characteristics for each port individually. It assumes that the release qualities will be linear combinations of the quality characteristics of each individual port. It solves the linear goal programming problem to find the optimal flows from the various ports and then uses the SELECT model again to recalculate the characteristics for the new set of flows. The process usually converges in two or three runs of both models. This approach is not suitable to investigate trade-offs among the water quality objectives, however.

The data for this application are the same used by Fontane and Schneider (1994). The quality profiles are presented in Figure 6.21. The selective withdrawal structure has five ports at elevations of 20ft, 60ft, 100ft, 140ft, and 180ft.

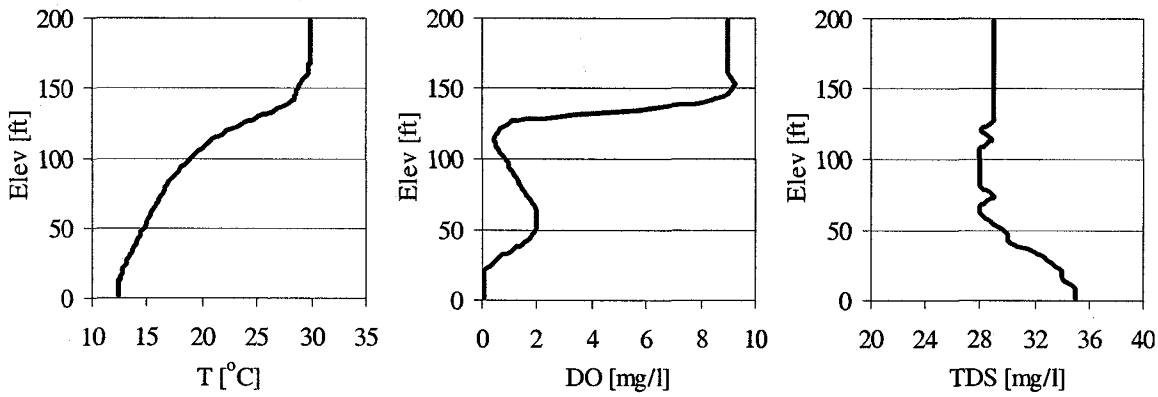


Figure 6.21 – Water Quality Profiles for the SELECT Model

Three test problems were constructed using the SELECT model: (i) minimize deviations from the targets of 16 °C for temperature and 8 mg/l for dissolved oxygen; (ii) minimize deviations from the targets of 14 °C for temperature, 9 mg/l for dissolved oxygen, and 28 mg/l for TDS; and (iii) minimize deviations from the targets of 22 °C for temperature, 7 mg/l for dissolved oxygen, and 30 mg/l for TDS, subject to the constraint that only two, but any two ports could be used. The quality targets were arbitrarily selected to create some conflicts among the objectives.

The higher dimensionality of the search space as well as the mathematical complexity of the SELECT model make these problems more difficult to solve than the ones related to the RESIMWP model, except for the Test Problem WEFR which is also hard from an optimization perspective. For instance, for the three problems using the SELECT model, the ε-NLP Solver had to be executed many times with various initial solutions and often failed to find a reasonable number of Pareto optimal solutions.

*Test Problem TD*

This problem has two objectives and five decision variables. The only constraint is that the sum of the flows through all ports must be equal to 1000 cfs.

Minimize $f_1(x_i)$, (6.13)

Minimize $f_2(x_i)$

where:

$$f_1(x_i) = [T(x_i) - 16]^2, \qquad f_2(x_i) = [DO(x_i) - 8]^2, \qquad i = 1, 2, \ldots, 5$$

subject to:

$$C_1 \equiv \sum_{i=1}^{5} x_i = 1000$$

$T(x_i)$ and $DO(x_i)$ are respectively the outflow temperature and dissolved oxygen computed by the SELECT model for the flow vector $x_i$, $i = 1, 2, \ldots, 5$. The equality constraint $C_1$, although a fairly simple constraint, makes this problem virtually unsolvable by multi-objective evolutionary algorithms. In most MOEA methods, including MOPSO and NSGA-II, the decision variables are randomly initialized and updated following some evolution rules which are partially random as well. It is virtually impossible to randomly generate five double-precision numbers that sum exactly to certain value. All solutions generated by the algorithms would be infeasible. The algorithms would try to gradually reduce the degree of infeasibility and eventually reach a feasible solution, but would never be able to find the whole Pareto optimal front. Moreover, crossover and mutation operators are also randomly implemented and would generate infeasible solutions as well. To deal with this situation, which is very common in real-world problems, a routine was developed and included in the MOPSO and NSGA-II Solvers, which is described in Chapter 4. The Pareto optimal front obtained using the ε-NLP Solver is shown in Figure 6.22. For some points, Excel's Solver could not verify optimality conditions.

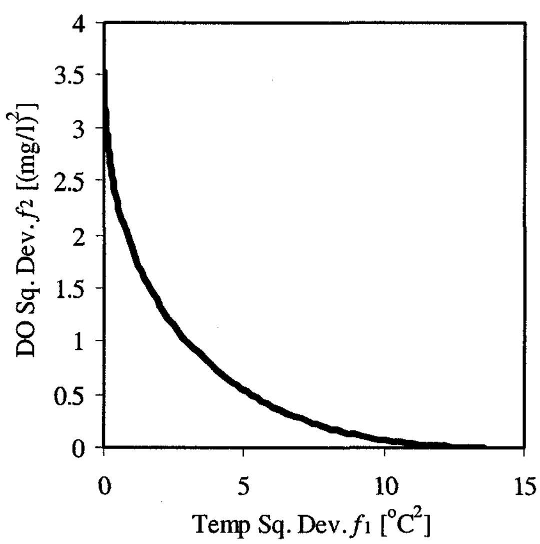



Figure 6.22 – Pareto Optimal Front for Test Problem TD

This problem was solved using a population size of 100 and 100 generations. The results are shown in Tables 6.70 to 6.76 and Figure 6.23. As discussed in section 4.1.1, the MOPSO post-processing routine (MOPSO-p) cannot be used when equality constraints are imposed on the decision variables, which is the case for Test Problems TD, TDT, and TDT2.

Table 6.70 – Generational Distance Metric for Test Problem TD

| Problem TD | Excel Add-ins | | | |
|---|---|---|---|---|
| | MOPSO | MOPSO-e | NSGA-II | RSearch |
| Best | 0.050847 | 0.052310 | 0.047820 | 0.074890 |
| Worst | 0.153746 | 0.067583 | 0.247424 | 0.322975 |
| Average | 0.068971 | **0.059593** | 0.086194 | 0.177630 |
| Median | 0.062031 | 0.059778 | 0.054522 | 0.174722 |
| Std.Dev. | 0.021017 | 0.003117 | 0.063385 | 0.067052 |

Table 6.71 – Spacing Metric for Test Problem TD

| Problem TD | Excel Add-ins | | | |
|---|---|---|---|---|
| | MOPSO | MOPSO-e | NSGA-II | RSearch |
| Best | 0.682635 | 0.631960 | 0.628462 | 0.637679 |
| Worst | 1.175527 | 1.132415 | 2.474544 | 1.919949 |
| Average | 0.920912 | **0.857347** | 0.871025 | 1.096944 |
| Median | 0.891787 | 0.842580 | 0.753179 | 1.004778 |
| Std.Dev. | 0.126928 | 0.121045 | 0.380301 | 0.336094 |

Table 6.72 – Inverted Generational Distance Metric for Test Problem TD

| Problem TD | Excel Add-ins | | | |
|---|---|---|---|---|
| | MOPSO | MOPSO-e | NSGA-II | RSearch |
| Best | 0.073055 | 0.075617 | 0.058478 | 0.087113 |
| Worst | 0.171819 | 0.139123 | 0.085718 | 0.220527 |
| Average | 0.107642 | 0.099814 | **0.067903** | 0.131022 |
| Median | 0.101058 | 0.096903 | 0.066698 | 0.126200 |
| Std.Dev. | 0.024209 | 0.016026 | 0.006942 | 0.035098 |

Table 6.73 – Diversity Metric for Test Problem TD

| Problem TD | Excel Add-ins | | | |
|---|---|---|---|---|
| | MOPSO | MOPSO-e | NSGA-II | RSearch |
| Best | 0.596437 | 0.575059 | 0.460345 | 0.539098 |
| Worst | 0.779778 | 0.746902 | 0.680566 | 0.723917 |
| Average | 0.677212 | 0.671118 | **0.544800** | 0.635923 |
| Median | 0.674994 | 0.664314 | 0.533470 | 0.632509 |
| Std.Dev. | 0.041136 | 0.042398 | 0.057751 | 0.043999 |

Table 6.74 – Processing Time for Test Problem TD

| Problem TD | Excel Add-ins | | | |
|---|---|---|---|---|
| | MOPSO | MOPSO-e | NSGA-II | RSearch |
| Best | 142.14 | 312.20 | 417.50 | 192.65 |
| Worst | 159.45 | 348.01 | 432.30 | 208.43 |
| Average | **151.83** | 323.43 | 420.38 | 196.14 |
| Median | 152.26 | 322.32 | 419.53 | 195.06 |
| Std.Dev. | 4.26 | 7.56 | 2.99 | 3.35 |

Table 6.75 – Dominated Ratio for Test Problem TD

| Problem TD | Excel Add-ins | | | |
|---|---|---|---|---|
| | MOPSO | MOPSO-e | NSGA-II | RSearch |
| Best | 0.09 | 0.07 | 0.04 | 0.78 |
| Worst | 0.27 | 0.23 | 0.19 | 0.91 |
| Average | 0.18 | 0.14 | **0.13** | 0.86 |
| Median | 0.18 | 0.15 | 0.12 | 0.86 |
| Std.Dev. | 0.05 | 0.04 | 0.03 | 0.03 |

Table 6.76 – Dominated Degree for Test Problem TD

| Problem TD | Excel Add-ins | | | |
|---|---|---|---|---|
| | MOPSO | MOPSO-e | NSGA-II | RSearch |
| Best | 0.16% | 0.14% | 0.12% | 0.82% |
| Worst | 0.98% | 0.93% | 3.06% | 1.24% |
| Average | 0.41% | **0.29%** | 0.56% | 1.01% |
| Median | 0.35% | 0.25% | 0.33% | 0.99% |
| Std.Dev. | 0.22% | 0.16% | 0.68% | 0.11% |

Analysis and Comparison

The results for the MOPSO and the NSGA-II Solvers were very similar for this test problem. MOPSO-e was a little better on GD and spacing while NSGA-II was slightly better on the iGD and diversity metrics. All methods eventually kept non-optimal

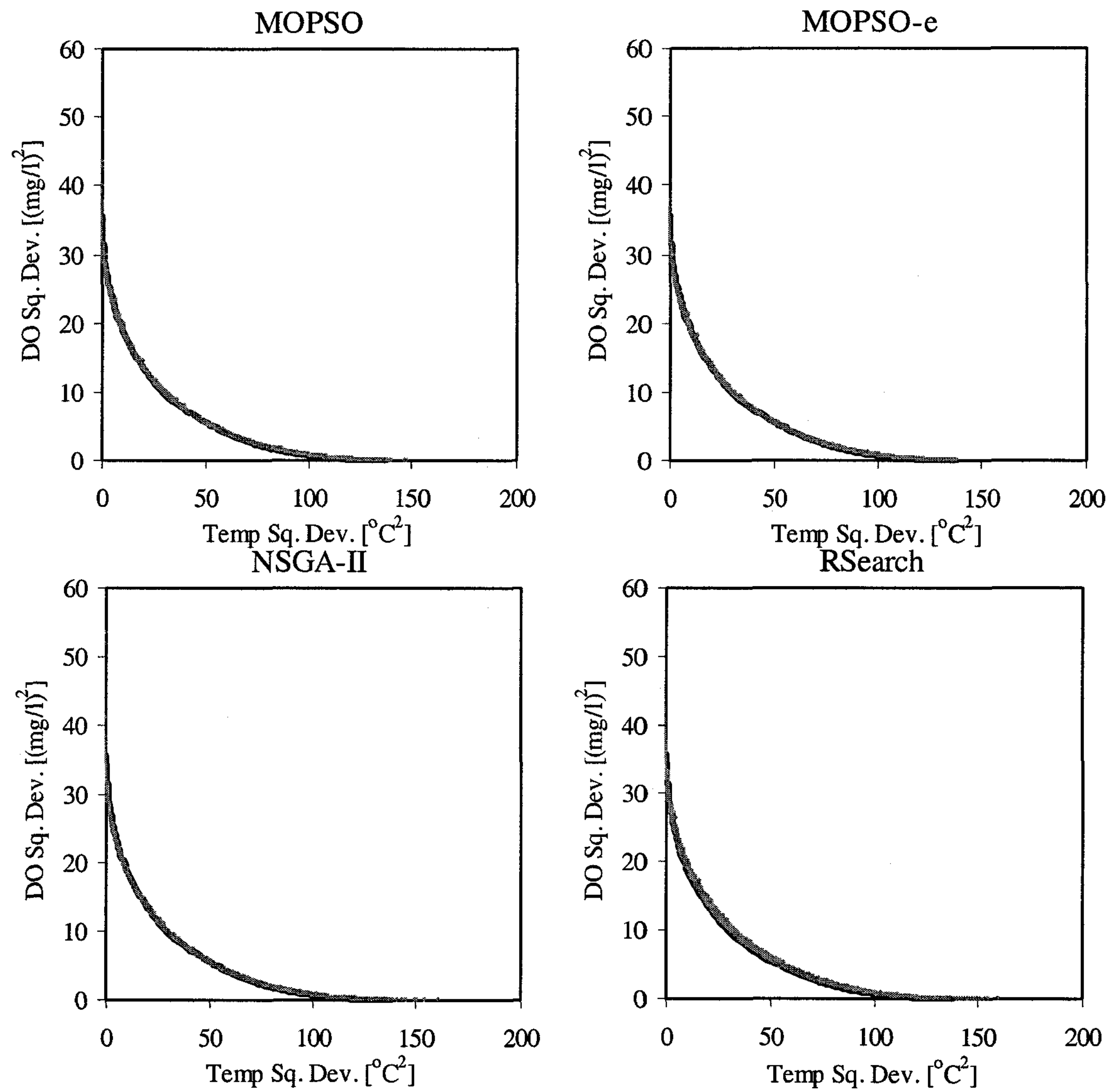



Figure 6.23 – Results for Test Problem TD

solutions on the extremities of the Pareto curve, but the dominated degrees were generally small. The MOPSO-e effectively addresses this problem (reflected on the reduced GD) at the expense of a worse coverage (reflected on increased iGD).

The average dominated ratio ranged from 0.13 for NSGA-II to 0.86 for the RSearch Solver, with average dominated degrees varying from 0.29% for MOPSO-e to about 1% for pure random search.

The evolutionary algorithms worked better compared to the pure random search approach. This is reasonable given the larger amplitude of the decision space (five decision variables varying from 0 to 1000) for this problem when compared to the previous ones.

The processing times for NSGA-II were considerably higher than for MOPSO. The processing time for this problem is dictated by the routine to deal with the equality constraint imposed on the decision variables, which is used within the crossover and mutation operators of NSGA-II and within the particle update procedure and the mutation operator of MOPSO. The difference in processing time is due to differences in the mutation approaches of the two solvers. MOPSO concentrates the mutations in the initial cycles rapidly reducing the number of particles subject to mutation. NSGA-II mutates its child solutions at the same rate in all generations, performing a much larger total number of mutations than MOPSO.

<u>Best Compromise Solutions</u>

The best compromise solutions for three different sets of weights are presented in Table 6.77 and Figure 6.24.

Table 6.77 – Best Compromise Solutions for Test Problem TD

| Case * | Flow [cfs] ** | T [°C] | DO [mg/l] |
|---|---|---|---|
| (a) Weight [1,1] | [446, 0, 0, 554, 0] | 21.8 | 5.0 |
| (b) Weight [1,0] | [ 43, 0, 0, 957, 0] | 16.0 | 2.0 |
| (c) Weight [0,1] | [868, 0, 0, 86, 46] | 27.7 | 8.0 |

* Weights: $[W_T, W_{DO}]$ ** Values between brackets are flows through Ports 1 to 5

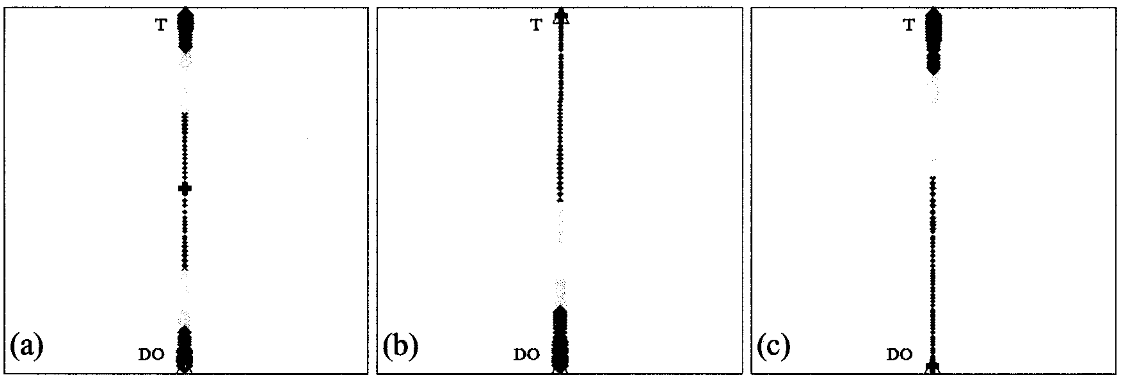



Figure 6.24 – Best Compromise Solutions for Test Problem TD

*Test Problem TDT*

This problem involves different targets and the additional objective of minimizing the square deviations from a TDS target of 28 mg/l.

Minimize $f_1(x_i)$, (6.14)

Minimize $f_2(x_i)$

Minimize $f_3(x_i)$

where:

$$f_1(x_i) = [T(x_i) - 14]^2, \qquad f_2(x_i) = [DO(x_i) - 9]^2, \qquad f_3(x_i) = [TDS(x_i) - 28]^2,$$

$i = 1, 2, \ldots, 5$

subject to:

$$C_1 \equiv \sum_{i=1}^{5} x_i = 1000$$

The targets were chosen to enhance conflicts among objectives. For meeting the temperature target, the two bottom ports must be used and any water taken from higher ports would cause larger deviations from the target. For dissolved oxygen, most water has to come from the top port, and just a little may be taken from the second highest. For total dissolved solids, the target cannot be exactly met but the best solution will occur when all water is taken from the middle port.

To solve this problem the ε-NLP Solver was executed many times, with different initial solutions, in order to get a reasonable number and spread of non-dominated solutions. For several solutions, the Excel Solver could not confirm optimality. The Pareto optimal surface is shown in Figure 6.25 and the ICC display in Figure 6.26.

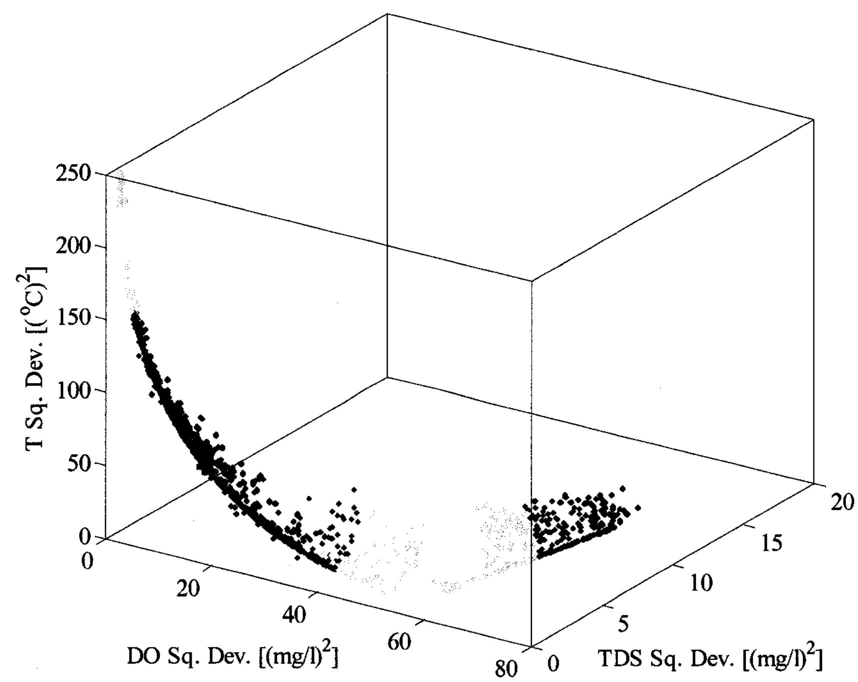



Figure 6.25 – Pareto Optimal Surface for Test Problem TDT

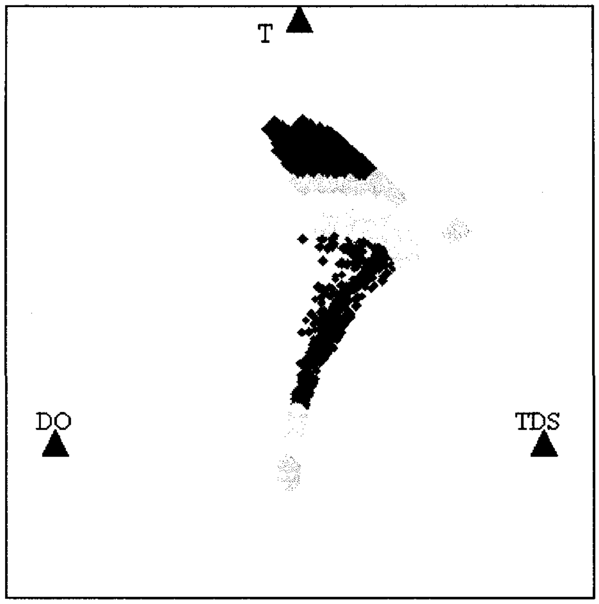



Figure 6.26 – ICC Graph of Pareto Optimal Set for Test Problem TDT

This problem was solved using a population size of 150 and 150 generations. The results are shown in Tables 6.78 to 6.83 and Figure 6.27.

Table 6.78 – Generational Distance Metric for Test Problem TDT

| Problem TDT | Excel Add-ins | | | |
|---|---|---|---|---|
| | MOPSO | MOPSO-e | NSGA-II | RSearch |
| Best | 0.378901 | 0.380384 | 0.146660 | 0.406292 |
| Worst | 0.473238 | 0.485112 | 0.265993 | 0.570258 |
| Average | 0.424000 | 0.431684 | **0.216546** | 0.483857 |
| Median | 0.423272 | 0.432213 | 0.220220 | 0.484713 |
| Std.Dev. | 0.025161 | 0.031764 | 0.028968 | 0.039502 |

Table 6.79 – Spacing Metric for Test Problem TDT

| Problem TDT | Excel Add-ins | | | |
|---|---|---|---|---|
| | MOPSO | MOPSO-e | NSGA-II | RSearch |
| Best | 2.081319 | 2.127334 | 1.686709 | 1.899009 |
| Worst | 2.773365 | 3.055745 | 3.300159 | 9.715416 |
| Average | **2.430528** | 2.472984 | 2.498304 | 3.739194 |
| Median | 2.424638 | 2.451747 | 2.431273 | 2.720312 |
| Std.Dev. | 0.167691 | 0.201799 | 0.347076 | 2.399345 |

Table 6.80 – Inverted Generational Distance Metric for Test Problem TDT

| Problem TDT | Excel Add-ins | | | |
|---|---|---|---|---|
| | MOPSO | MOPSO-e | NSGA-II | RSearch |
| Best | 0.038101 | 0.038094 | 0.042104 | 0.119317 |
| Worst | 0.050938 | 0.050581 | 0.060787 | 0.253444 |
| Average | **0.042546** | 0.043361 | 0.049452 | 0.181114 |
| Median | 0.042483 | 0.043025 | 0.049130 | 0.175305 |
| Std.Dev. | 0.003021 | 0.002804 | 0.004352 | 0.038127 |

Table 6.81 – Processing Time for Test Problem TDT

| Problem TDT | Excel Add-ins | | | |
|---|---|---|---|---|
| | MOPSO | MOPSO-e | NSGA-II | RSearch |
| Best | 872.19 | 1605.88 | 868.73 | 303.70 |
| Worst | 912.20 | 1682.99 | 968.30 | 347.52 |
| Average | 888.73 | 1646.39 | 913.42 | **318.45** |
| Median | 890.55 | 1650.17 | 920.86 | 314.96 |
| Std.Dev. | 10.26 | 20.86 | 29.51 | 9.44 |

Table 6.82 – Dominated Ratio for Test Problem TDT

| Problem TDT | Excel Add-ins | | | |
|---|---|---|---|---|
| | MOPSO | MOPSO-e | NSGA-II | RSearch |
| Best | 0.86 | 0.69 | 0.77 | 0.98 |
| Worst | 0.96 | 0.83 | 0.96 | 1.00 |
| Average | 0.92 | **0.76** | 0.90 | 0.99 |
| Median | 0.93 | 0.76 | 0.93 | 0.99 |
| Std.Dev. | 0.02 | 0.03 | 0.05 | 0.01 |

Table 6.83 – Dominated Degree for Test Problem TDT

| Problem TDT | Excel Add-ins | | | |
|---|---|---|---|---|
| | MOPSO | MOPSO-e | NSGA-II | RSearch |
| Best | 3.52% | 4.11% | 6.98% | 8.93% |
| Worst | 6.30% | 7.47% | 11.23% | 11.79% |
| Average | **4.88%** | 5.75% | 9.16% | 10.15% |
| Median | 5.00% | 5.75% | 9.31% | 10.06% |
| Std.Dev. | 0.77% | 0.88% | 0.92% | 0.73% |

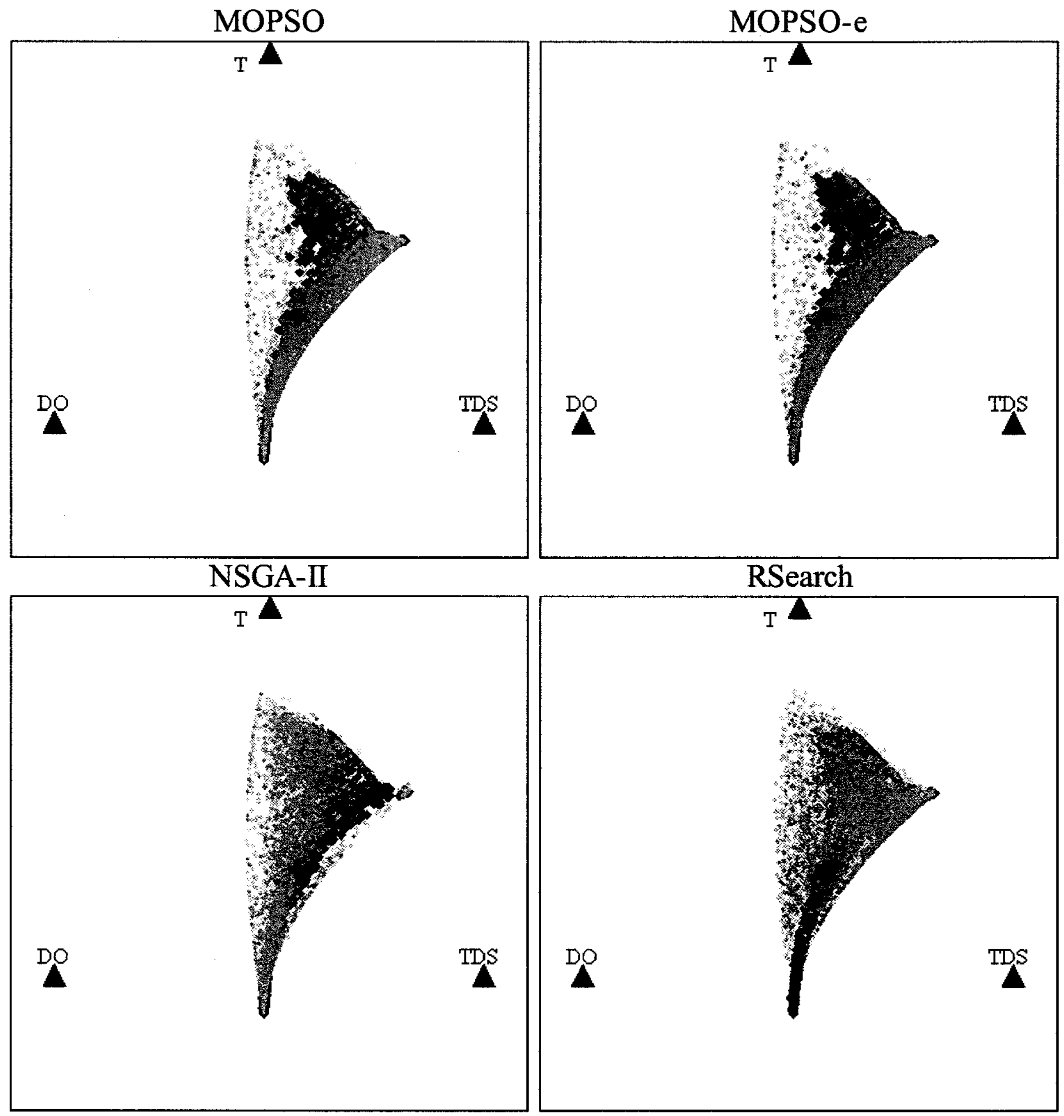



Figure 6.27 – Results for Test Problem TDT

Analysis and Comparison

This test problem offered an interesting challenge to the evolutionary algorithms. Total dissolved solids show very little variation on the upper layers of the reservoir (from 200 ft to about 50 ft) increasing rapidly when closer to the bottom (below 50 ft). The target for TDS was set to 28 mg/l, which is the absolute minimum value on the TDS profile, and it occurs only on the middle layers (around 100 ft). The best solutions to meet this target will occur if most water in taken from the third port, which is located exactly at 100 ft elevation. If water is taken from any port but the lowest one (located at 20 ft) the deviations from the TDS target will not be large however. This increases the likelihood of generating solutions that are close to the TDS target, creating an uneven distribution of solutions.

The effect of this uneven distribution on MOPSO and NSGA-II is completely different however. Both algorithms failed to produce a well distributed set of non-dominated solutions (see Figure 6.27). For MOPSO, and MOPSO-e as well, the solutions are concentrated exactly on the region of small deviations from the TDS target. This behavior can be explained by the way the algorithm proceeds in the search. The probability that a particle will be selected as a global driver for others is inversely related to the density of solutions around that particle, but if there are many solutions concentrated in one region the probability that any of them will be selected is still high. This makes the MOPSO algorithm favor the search on that region, creating a very strong clustering effect. This problem was not detected by the spacing metric, and it is actually an example where this metric can be misleading (see discussion in section 3.3).

The clustering effect in NSGA-II seems to be smaller but occurs in a different region. The explanation for that behavior is less straightforward, and it is related to the crowding distance operator. To investigate this matter, intermediate generations in a NSGA-II run for this problem were recorded, and are shown in Figure 6.28.

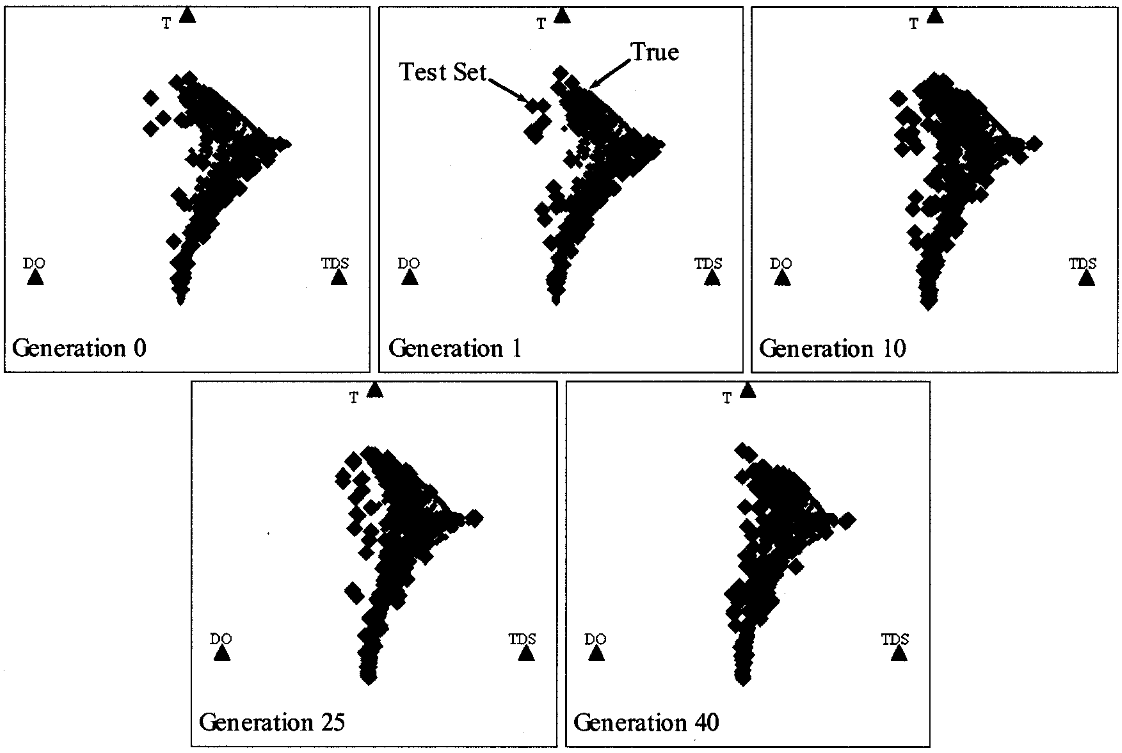


Figure 6.28 – NSGA-II Intermediate Generations for Test Problem TDT

In the beginning (generations 0 and 1), as it should be expected, the solutions are concentrated exactly in the same region as the MOPSO solutions, i.e. where the small deviations from the TDS target are located. As the search evolves, the solutions moved from that region to others where the variations from the TDS target are larger. This behavior may be explained by the way the crowding operator works. In every generation, NSGA-II combines the parent and child populations, and the merged population is then ranked. What usually happens, especially after a few generations, is that the number of solutions ranked first in the merged population is larger than the original population size.

The crowding distance is then used to select solutions among those rank-one solutions to form the parent population for the next generation. The solutions crowded on the region of small deviations from the TDS target are progressively eliminated by the crowding distance operator. This effect does not happen, for example, from generation zero to one because at generation one the rank-one population is smaller than the population size and thus the crowding operator is not used.

This is an important issue for further research, that is, how to avoid this type of clustering effect on both the MOPSO and NSGA-II algorithms.

The NSGA-II solutions presented better GD metrics than MOPSO and MOPSO-e solutions, with worse dominated degrees. However, this seems to be a contradiction since GD and DD are usually directly correlated. Since the dominated degree is the average (across objectives) difference in the objective values between the dominated solution in the test set and the corresponding non-dominated solution in the true Pareto set, a very large difference in one objective will increase significantly the dominated degree. When NSGA-II favors solutions with higher deviations from the TDS target, many of them will be dominated by others with much smaller TDS deviations, increasing the values of the overall dominated degrees. Therefore, the NSGA-II apparently high dominated degrees are actually a consequence of the clustering pattern as well.

Best Compromise Solutions

The best compromise solutions for four different sets of weights are presented in Table 6.84 and Figure 6.29.

Table 6.84 – Best Compromise Solutions for Test Problem TDT

| Case * | Flow [cfs] ** | T [°C] | DO [mg/l] | TDS [mg/l] |
|---|---|---|---|---|
| (a) Weight [1,1,1] | [439, 1, 1, 559, 0] | 21.7 | 5.0 | 28.9 |
| (b) Weight [1,0,0] | [0, 0, 0, 405, 595] | 14.0 | 1.0 | 31.4 |
| (c) Weight [0,1,0] | [998, 0, 2, 0, 0] | 29.7 | 9.0 | 29.0 |
| (d) Weight [0,0,1] | [12, 1, 1, 986, 0] | 19.3 | 0.8 | 28.2 |

* Weights: [$W_T$,$W_{DO}$,$W_{TDS}$] ** Values between brackets are flows through Ports 1 to 5

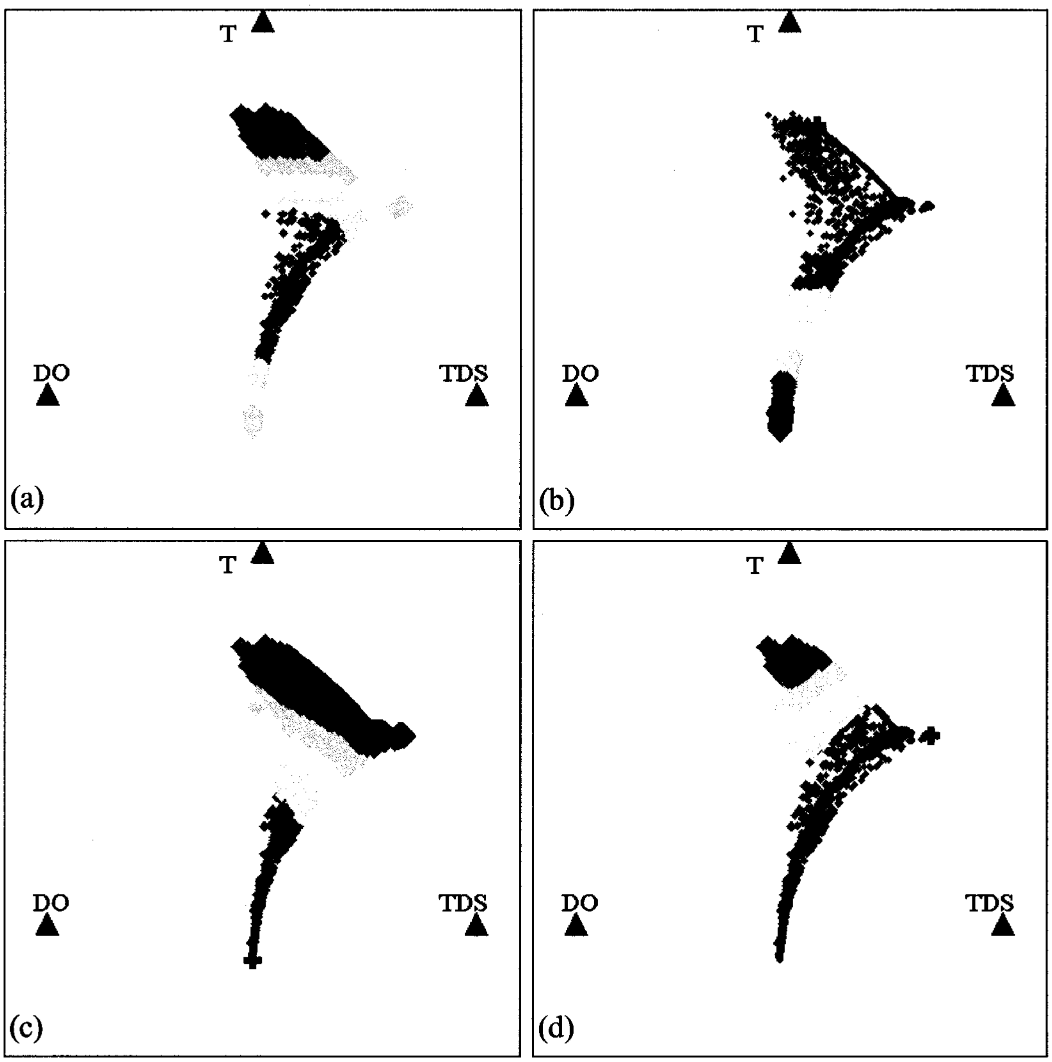



Figure 6.29 – Best Compromise Solutions for Test Problem TDT

*Test Problem TDT2*

A variation of the Test Problem TDT is used here with targets of 22 °C for temperature, 7 mg/l for dissolved oxygen, and 30 mg/l for total dissolved solids. The solution of this new three objective problem is shown in Figures 6.30 and 6.31. Most Pareto optimal solutions obtained for these targets involve at least three different ports.

The Test Problem TDT2 is built by restricting the total number of ports to be used to two. This would be important, for example, in a planning or design exercise if only two ports are to be constructed, out of five alternative depths. The multi-objective analysis would help the decision making on what depths the two ports should be selected.

The optimization problem is set up by adding five binary variables $b_i$, which would determine if port $i$ is to be used ($b_i = 1$) or not ($b_i = 0$), and a constraint that forces the sum of such variables to be equal to two. The mathematical definition of this problem is given in Eq. 6.15 to follow.

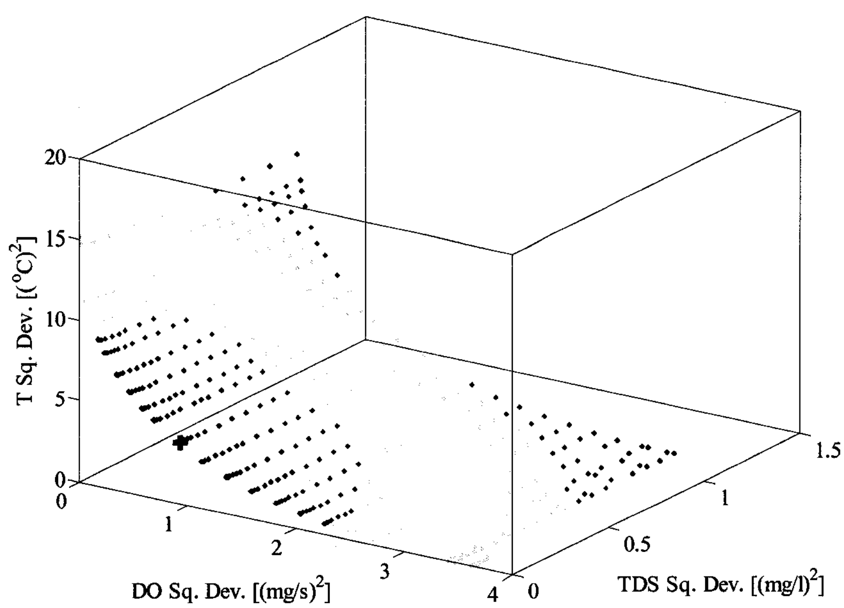



Figure 6.30 – Pareto Optimal Surface for Test Problem TDT2 without Binary Constraints

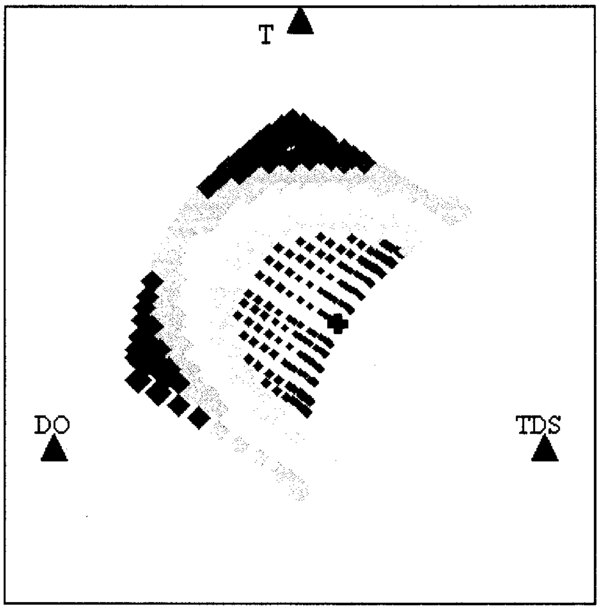



Figure 6.31 – ICC Graph for Test Problem TDT2 without Binary Constraints

Minimize $f_1(b_i{\cdot}x_i)$, (6.15)

Minimize $f_2(b_i{\cdot}x_i)$

Minimize $f_3(b_i{\cdot}x_i)$

where:

$$f_1(x_i)=\left[T(b_i\cdot x_i)-22\right]^2,\quad f_2(x_i)=\left[DO(b_i\cdot x_i)-7\right]^2,\quad f_3(x_i)=\left[TDS(b_i\cdot x_i)-30\right]^2,$$

$i = 1, 2, \ldots, 5$

subject to:

$$C_1 \equiv \sum_{i=1}^{5} x_i = 1000$$

$$C_2 \equiv \sum_{i=1}^{5} b_i = 2,\qquad b_i \text{ binary},\qquad i = 1, 2, \ldots, 5$$

This problem is considerably more complex and difficult to solve. The binary variables introduce a number of discontinuities, making the problem extremely difficult to solve with gradient-based search techniques. In fact, even with a large number of trials with different initial solutions and search parameters, the ε-NLP Solver could not identify the possible ranges for each objective, necessary to define the epsilon bounds.

The following procedure was used to find the Pareto optimal set shown in Figure 6.32. The MOPSO, NSGA-II, and RSearch Solvers were used to generate thirty different sets of 150 non-dominated solutions. All solutions were merged, over thirteen thousand, and the dominated ones were excluded. This final set was then tested for possible Pareto improvements using Excel Solver. The solver now could be more successfully applied since the starting solutions should be at least near optimal. Even so, for many solutions, optimality conditions could not be verified by the Excel Solver. Moreover, it is very difficult to confirm if the obtained Pareto optimal set is covering the whole extension of the possible trade-offs.

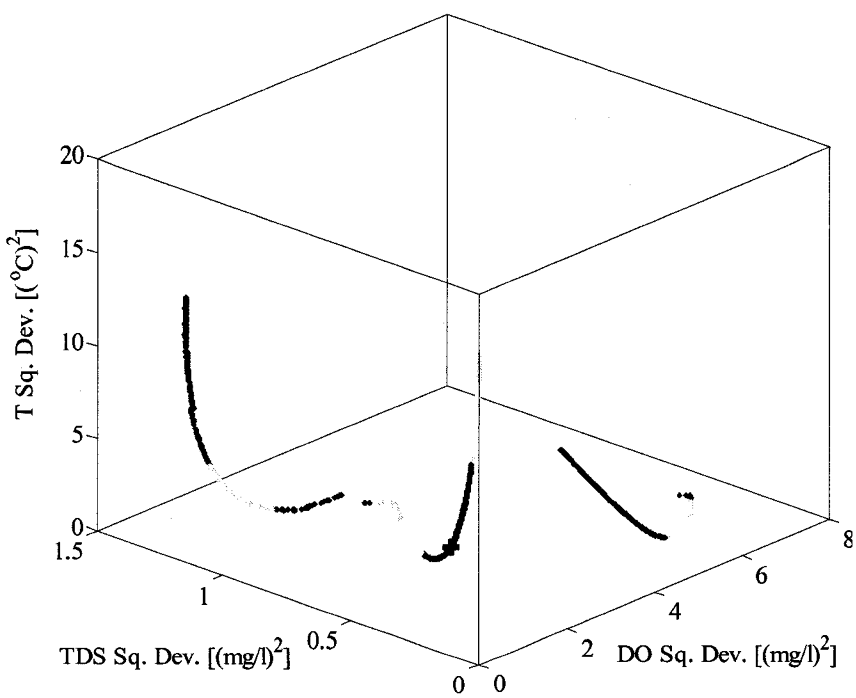



Figure 6.32 – Pareto Optimal Surface for Test Problem TDT2

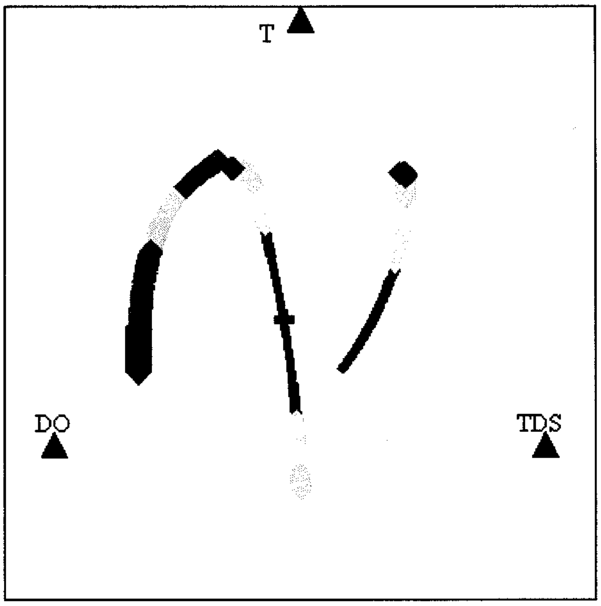



Figure 6.33 – ICC Graph of Pareto Optimal Set for Test Problem TDT2

As one can see from Figure 6.32, the Pareto optimal set is formed by three disconnected three-dimensional curves, each one corresponding to a combination of two ports. The first segment is formed by solutions that use Port 1 and Port 4, the second group uses Port 1 and Port 5, and the third uses Port 2 and Port 5. Figure 6.34 shows an ICC graph displaying the three different groups of solutions.

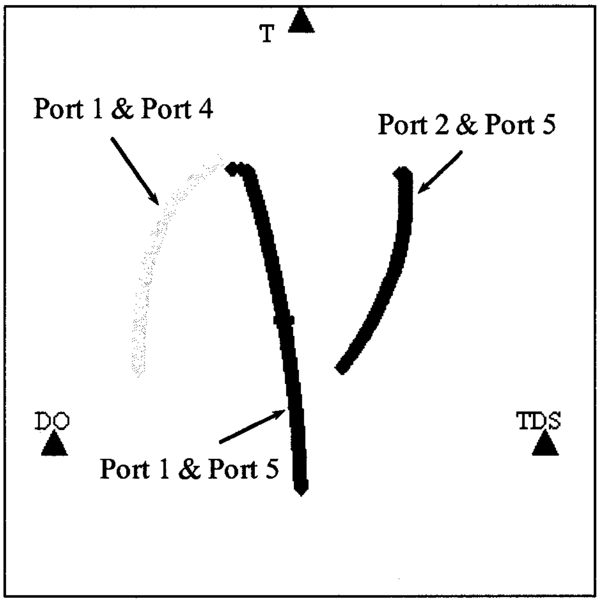



Figure 6.34 – Different Subsets of Solutions for Test Problem TDT2

This problem was solved using a population size of 100 and 100 generations. The results are shown in Tables 6.85 to 6.92 and Figure 6.35. Due to long processing times, the MOPSO with crossover (MOPSO-e) was not run for this problem. Pair-wise comparisons on the average dominated ratio and dominated degree are also presented for this problem given the difficulty in identifying the true Pareto optimal set.

Table 6.85 – Generational Distance Metric for Test Problem TDT2

| Problem TDT2 | Excel Add-ins | | |
|---|---|---|---|
| | MOPSO | NSGA-II | RSearch |
| Best | 0.000150 | 0.000028 | 0.009667 |
| Worst | 0.467266 | 0.809315 | 0.943610 |
| Average | **0.054382** | 0.096709 | 0.495260 |
| Median | 0.001074 | 0.000159 | 0.538024 |
| Std.Dev. | 0.120863 | 0.202233 | 0.247741 |

Table 6.86 – Spacing Metric for Test Problem TDT2

| Problem TDT2 | Excel Add-ins | | |
|---|---|---|---|
| | MOPSO | NSGA-II | RSearch |
| Best | 0.210048 | 0.175571 | 0.269964 |
| Worst | 6.571078 | 0.401949 | 5.480728 |
| Average | 0.605054 | **0.301390** | 2.757810 |
| Median | 0.290169 | 0.298644 | 1.658270 |
| Std.Dev. | 1.182440 | 0.060294 | 2.044514 |

Table 6.87 – Inverted Generational Distance Metric for Test Problem TDT2

| Problem TDT2 | Excel Add-ins | | |
|---|---|---|---|
| | MOPSO | NSGA-II | RSearch |
| Best | 0.002697 | 0.003779 | 0.004320 |
| Worst | 0.004885 | 0.009318 | 0.032194 |
| Average | **0.003661** | 0.005848 | 0.015263 |
| Median | 0.003597 | 0.005653 | 0.013737 |
| Std.Dev. | 0.000554 | 0.001590 | 0.008269 |

Table 6.88 – Processing Time for Test Problem TDT2

| Problem TDT2 | Excel Add-ins | | |
|---|---|---|---|
| | MOPSO | NSGA-II | RSearch |
| Best | 4982.44 | 9237.67 | 11659.52 |
| Worst | 11822.00 | 11830.51 | 15675.10 |
| Average | **6882.84** | 10478.05 | 13316.55 |
| Median | 6986.15 | 10469.44 | 12963.88 |
| Std.Dev. | 1370.37 | 676.01 | 1194.28 |

Table 6.89 – Dominated Ratio for Test Problem TDT2

| Problem TDT2 | Excel Add-ins | | |
|---|---|---|---|
| | MOPSO | NSGA-II | RSearch |
| Best | 0.07 | 0.08 | 0.03 |
| Worst | 0.20 | 0.21 | 0.14 |
| Average | 0.12 | 0.14 | **0.08** |
| Median | 0.12 | 0.14 | 0.07 |
| Std.Dev. | 0.03 | 0.03 | 0.03 |

Table 6.90 – Dominated Degree for Test Problem TDT2

| Problem TDT2 | Excel Add-ins | | |
|---|---|---|---|
| | MOPSO | NSGA-II | RSearch |
| Best | 0.00% | 0.00% | 16.28% |
| Worst | 37.46% | 36.84% | 130.14% |
| Average | **3.82%** | 5.71% | 67.07% |
| Median | 0.34% | 0.01% | 68.93% |
| Std.Dev. | 8.69% | 9.75% | 27.83% |

Table 6.91 – Pair-wise Comparison on Average DR for Test Problem TDT2

| P. WEFR* | MOPSO | NSGA-II | RSearch |
|---|---|---|---|
| MOPSO | - | 0.0101 | 0.0071 |
| NSGA-II | 0.0017 | - | 0.0053 |
| RSearch | 0.0107 | 0.0523 | - |

* The element in the first row second column, 0.0101, means that about 1% of the solutions in the MOPSO test sets are dominated by solutions in the NSGA-II test sets.

Table 6.92 – Pair-wise Comparison on Average DD for Test Problem TDT2

| P. WEFR* | MOPSO | NSGA-II | RSearch |
|---|---|---|---|
| MOPSO | - | 4.81% | 0.97% |
| NSGA-II | 3.07% | - | 3.54% |
| RSearch | 11.22% | 24.49% | - |

* The element in the first row second column, 4.81%, means that for those solutions in the MOPSO test sets that are dominated by solutions in the NSGA-II test sets, the average degree of domination is about 5% of the ranges of the objective functions.

Analysis and Comparison

MOPSO had better results than NSGA-II in terms of generational distance, inverted generational distance, and dominated degree, while NSGA-II was better in the spacing metric. The evolutionary algorithms generally outperformed the pure random search, with the exception of the dominated ratio, where RSearch was slightly better. All these metrics must be carefully analysed since they may be influenced by the way the "true" Pareto optimal set was obtained, where the solutions of the three methods were used as initial solutions for a nonlinear optimization search using the Excel Solver.

The pair-wise comparisons between the methods confirm that the evolutionary algorithms produce much better results than the pure random search approach. NSGA-II solutions dominate MOPSO solutions more frequently than vice versa, and MOPSO solutions present slightly higher dominated degree than NSGA-II. The overall results, however, suggest a rather equal performance for the two evolutionary algorithms.

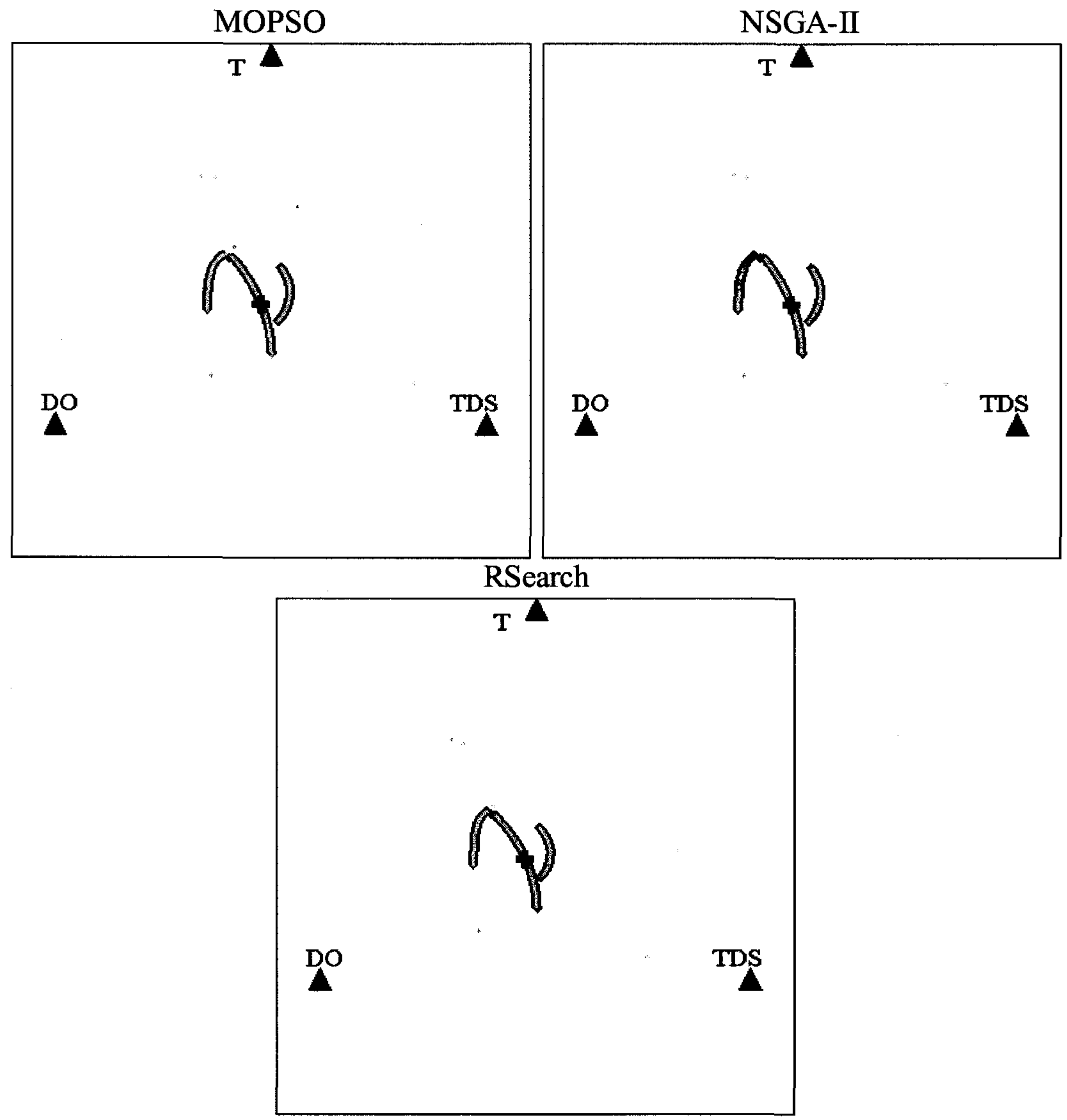



Figure 6.35 – Results for Test Problem TDT2

Best Compromise Solutions

The best compromise solutions for four different sets of weights are presented in Table 6.93 and Figure 6.36.

Table 6.93 – Best Compromise Solutions for Test Problem TDT2

| Case * | Flow [cfs] ** | T [°C] | DO [mg/l] | TDS [mg/l] |
|---|---|---|---|---|
| (a) Weight [1,1,1] | [660, 0, 0, 0, 340] | 24.1 | 6.1 | 30.5 |
| (b) Weight [1,0,0] | [536, 0, 0, 0, 464] | 22.0 | 5.0 | 31.0 |
| (c) Weight [0,1,0] | [768, 0, 0, 0, 232] | 25.9 | 7.0 | 30.1 |
| (d) Weight [0,0,1] | [782, 0, 0, 0, 218] | 26.1 | 7.1 | 30.0 |

* Weights: [$W_T$, $W_{DO}$, $W_{TDS}$] ** Values between brackets are flows through Ports 1 to 5

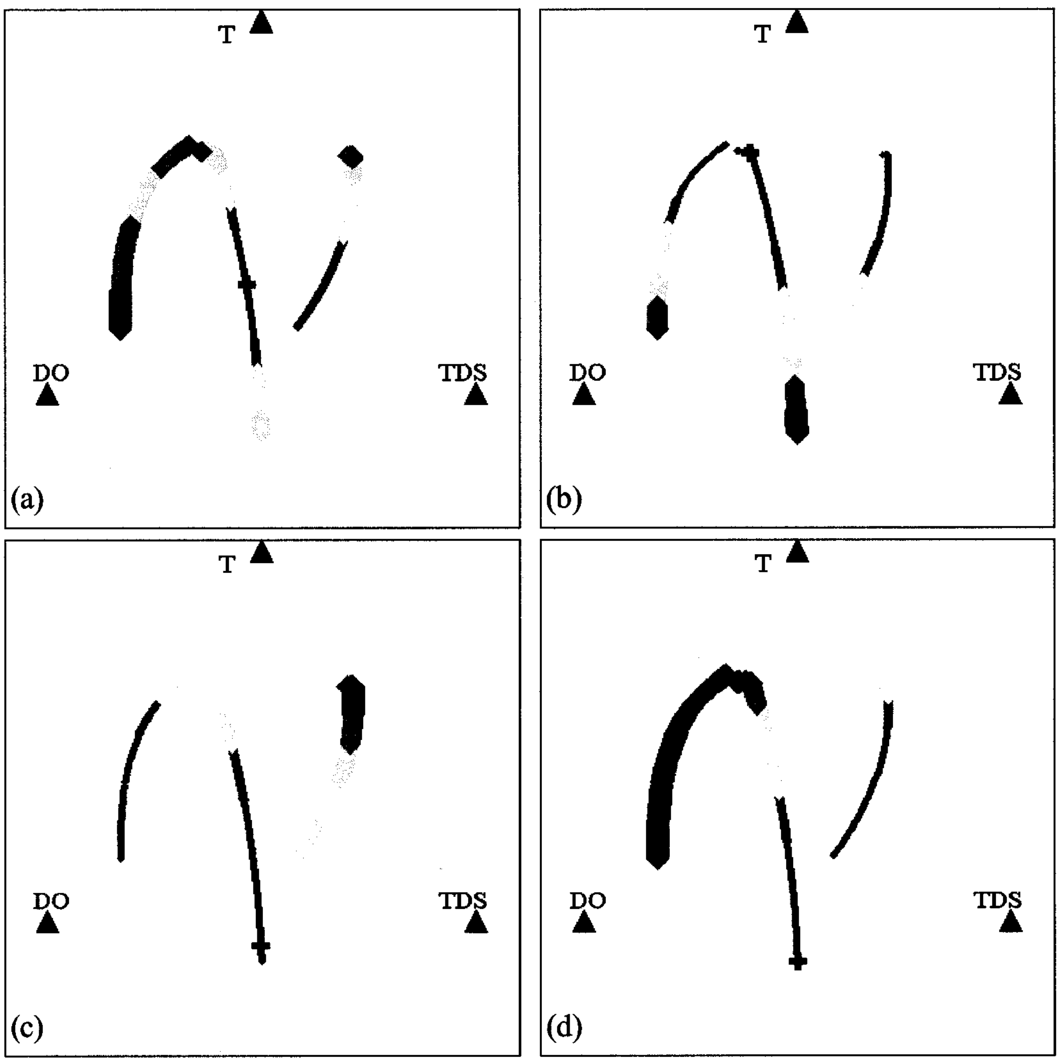



Figure 6.36 – Best Compromise Solutions for Test Problem TDT2

5.2.3. Multi-Purpose Rule-Based Reservoir Operation with Fuzzy Objectives

This application is based on the work by Fontane et al. (1997) and Moncada (1999), where fuzzy membership functions were derived and applied to the operational planning of the Grey Mountain Reservoir Project on the Cache La Poudre River, Colorado. Although studied and planned for many years this reservoir has never been built.

The Grey Mountain reservoir was planned to be located in the main stem of the Cache La Poudre River, approximately ten miles northwest of the city of Fort Collins, Colorado (Figure 6.37). It would have a storage capacity of about 241 Mcm (195,000 acre-ft), with 229 Mcm of active volume and 12 Mcm of flood space, and the main purposes of: (i) municipal and industrial water supply, (ii) energy production, (iii) flood control, (iv) recreational use, and (v) fish habitat needs.

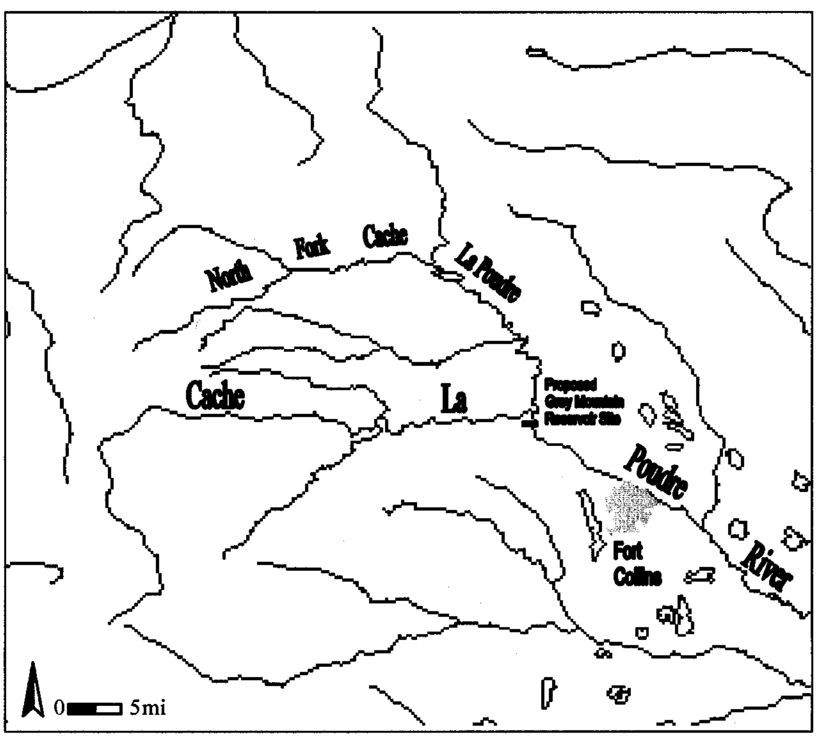



Figure 6.37 – Proposed Grey Mountain Reservoir Site

Moncada (1999) addressed the reservoir operation planning as a fuzzy optimization problem, with a fuzzy goal and seven objectives defined as fuzzy constraints. The fuzzy goal was related to the assertion that the stored volume at the end of the water year (end of September) would be in the neighborhood of the total active volume. The fuzzy constraints were defined as: (i) *Adequate* municipal and industrial water supply, (ii) *Dependable* flood control space, (iii) *Efficient* hydropower, (iv) *Enjoyable* rafting, (v) *Enjoyable* kayaking, (vi) *Adequate* angling, and (vii) *Suitable* fish habitat. The author derived membership functions for the fuzzy goal and constraints based on interviews with water resources experts and water users, and used dynamic programming to find the set of optimal monthly releases that would maximize the weighted sum of the fuzzy goal and all the fuzzy constraints.

Optimal release policies reflect the best possible reservoir operation under the (very strong) assumption of perfect knowledge of future inflows. If the actual inflows differ substantially from the assumed values, catastrophic failures could eventually happen (serious shortages or floods). That is why reservoirs operators are usually reluctant to follow optimal release policies.

A different approach is followed here. Instead of searching for optimal releases, multi-objective optimization is used to find a set of Pareto optimal end-of-the-month storage guide curves, using the same membership functions derived by Moncada (1999).

Guide curves are often preferred in real-world reservoir operations since they provide a consistent operational rule, which could avoid potentially serious failures.

The rule-based reservoir simulation model was implemented by modifying the RESIMWP model described in section 6.2.1 to perform the water balance in the following way:

$$V(t+1) = V(t) + Inflow(t) - \mathrm{Re}l(t) \tag{6.16}$$

$$\mathrm{Re}l(t) = \min\{R_{Rule}(t)\ ,\ D^*(t)\} \tag{6.17}$$

$$R_{Rule}(t) = \max\{[V(t) + Inflow(t) - GC_t]\ ,\ 0\} \tag{6.18}$$

$$D^* = \max\{[D_{SR}(t) + D_{WS}(t)]\ , D_{FC}(t)\ , D_{HP}(t)\ , D_R(t)\ , D_A(t)\ , D_{FH}(t)\} \tag{6.19}$$

where: $GC_t$ is the guide curve value for month $t$, $D_{SR}$ is the demand associated with downstream senior water rights, $D_{WS}$ is the demand for municipal and industrial water supply, $D_{FC}$ is the release necessary to ensure that 100% of the flood space is available, $D_{HP}$ is the water release necessary to supply the energy demand, $D_R$ is the release to ensure ideal conditions for rafting, $D_A$ is the release for ideal angling conditions, $D_{FH}$ is the release for ideal fish habitat support. All the demands were taken from Moncada (1999). The senior water rights, municipal and industrial water supply, and hydropower demands were adjusted in order to allow a richer analysis of trade-offs. In fact, these demands were so high that most of the time their associated membership functions would be simply zero, since they were defined as functions of the percentage of the demand being met. Evaporation and seepage rates were not available. Minimum environmental flows are assumed to be included as a senior water right, i.e. they can not be diverted to fulfill the water supply objective (the only consumptive demand being analyzed).

The model equations reflect the following operational rule: (i) release the maximum demand if the volume in the beginning of the month plus the inflow is sufficient to supply the maximum demand and to keep the volume at the end of the month above or equal to the guide curve value; (ii) release as much as possible while keeping the volume at the end of the month exactly at the guide curve value; or (iii) do not release any water if the volume in the beginning of the month plus the inflow is less than the guide curve value.

*Test Problem FUZZY*

This test problem includes six of the seven fuzzy objectives considered by Moncada (1999). Given that the kayaking and the rafting membership functions are very similar, only the rafting was included. The optimization problem was formulated as follows:

$$\text{Maximize } \{U_{WS}(GC_t),\ U_{FC}(GC_t),\ U_{HD}(GC_t),\ U_R(GC_t),\ U_A(GC_t),\ U_{FH}(GC_t)\} \qquad (6.20)$$

where: $t = 1\ldots12$, and $U_{WS}$, $U_{FC}$, $U_{HD}$, $U_R$, $U_A$, and $U_{FH}$ are the average values of the membership functions derived by Moncada (1999) for municipal and industrial water supply, flood control, hydropower, rafting, angling, and fish habitat support, respectively.

The month with the minimum value in the guide curve was also included in the optimization problem as a decision variable; an integer variable from 5 to 8 representing a month of the water year (February to May). After this month is set, the lower and upper bounds for each value of the guide curve are calculated in the spreadsheet, in such way that the guide curve values will decrease from month 1 (October) until the minimum month and then increase from there to month 12 (September). This is automatically done through formulas in the spreadsheet. The bounds are entered in the solvers' interface as cell references in the spreadsheet rather than fixed values. As the decision variables are sequentially defined by the algorithms, the bounds are dynamically updated ensuring that the guide curve will have the desired smooth shape.

The MOPSO, NSGA-II, and RSearch Solvers were run thirty times each, generating a total of nine thousand different solutions, i.e. 9000 guide curves and their associated objective values. These solutions were all merged and the dominated ones were discarded. The resulting non-dominated solutions are presented in the ICC graph shown in Figure 6.38.

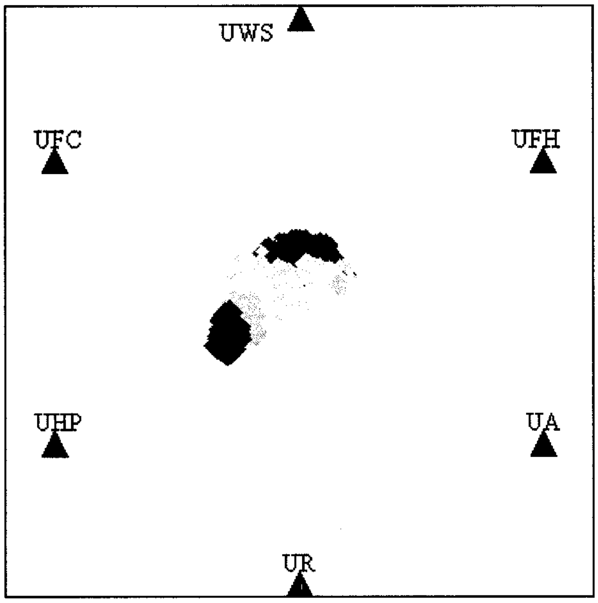



Figure 6.38 – ICC Graph of Non-dominated Solutions for Test Problem FUZZY

This problem was run with a population size of 100 and 100 generations. A few attempts to use the ε-NLP solver produced very poor results due to the size of the problem (in terms of objective functions) and the complexity introduced by allowing the minimum month of the guide curve be a decision variable as well. For this problem then, only pair-wise comparisons between MOPSO, NSGA-II, and RSearch on the dominated ratio and dominated degree metrics are presented in Tables 6.94 and 6.95.

Figure 6.39 shows the ICC graphs for comparison of the non-dominated set obtained by merging all solutions from the three solvers and the individual results of each solver.

Table 6.94 – Pair-wise Comparison on Average DR for Test Problem FUZZY

| P. WEFR* | MOPSO | NSGA-II | RSearch |
|---|---|---|---|
| MOPSO | - | 0.0240 | 0.0116 |
| NSGA-II | 0.0107 | - | 0.0603 |
| RSearch | 0.1611 | 0.1427 | - |

* The element in the first row second column, 0.0240, means that about 2.5% of the solutions in the MOPSO test sets are dominated by solutions in the NSGA-II test sets.

Table 6.95 – Pair-wise Comparison on Average DD for Test Problem FUZZY

| P. WEFR* | MOPSO | NSGA-II | RSearch |
|---|---|---|---|
| MOPSO | - | 0.74% | 1.60% |
| NSGA-II | 6.66% | - | 4.77% |
| RSearch | 12.66% | 10.70% | - |

* The element in the first row second column, 0.74%, means that for those solutions in the MOPSO test sets that are dominated by solutions in the NSGA-II test sets, the average degree of domination is about 0.7% of the ranges of the objective functions.

Analysis and Comparison

The metrics presented in Tables 6.94 and 6.95 can be misleading. The dominated ratio and the dominated degree metrics measure only the convergence performance. On those metrics, MOPSO seems to be performing the best. Analyzing Figure 6.39, one can easily notice that MOPSO had a very poor coverage of the Pareto optimal set (or at least the best available approximation of the Pareto optimal set). The MOPSO solutions are very concentrated, probably because the applied search coefficients $c_1$ and $c_2$ (equal to one) were too small for this problem. MOPSO performed a very refined local search but explored just a small fraction of the Pareto optimal set. This is compatible with the observed values for the dominated ratios and the dominated degrees.

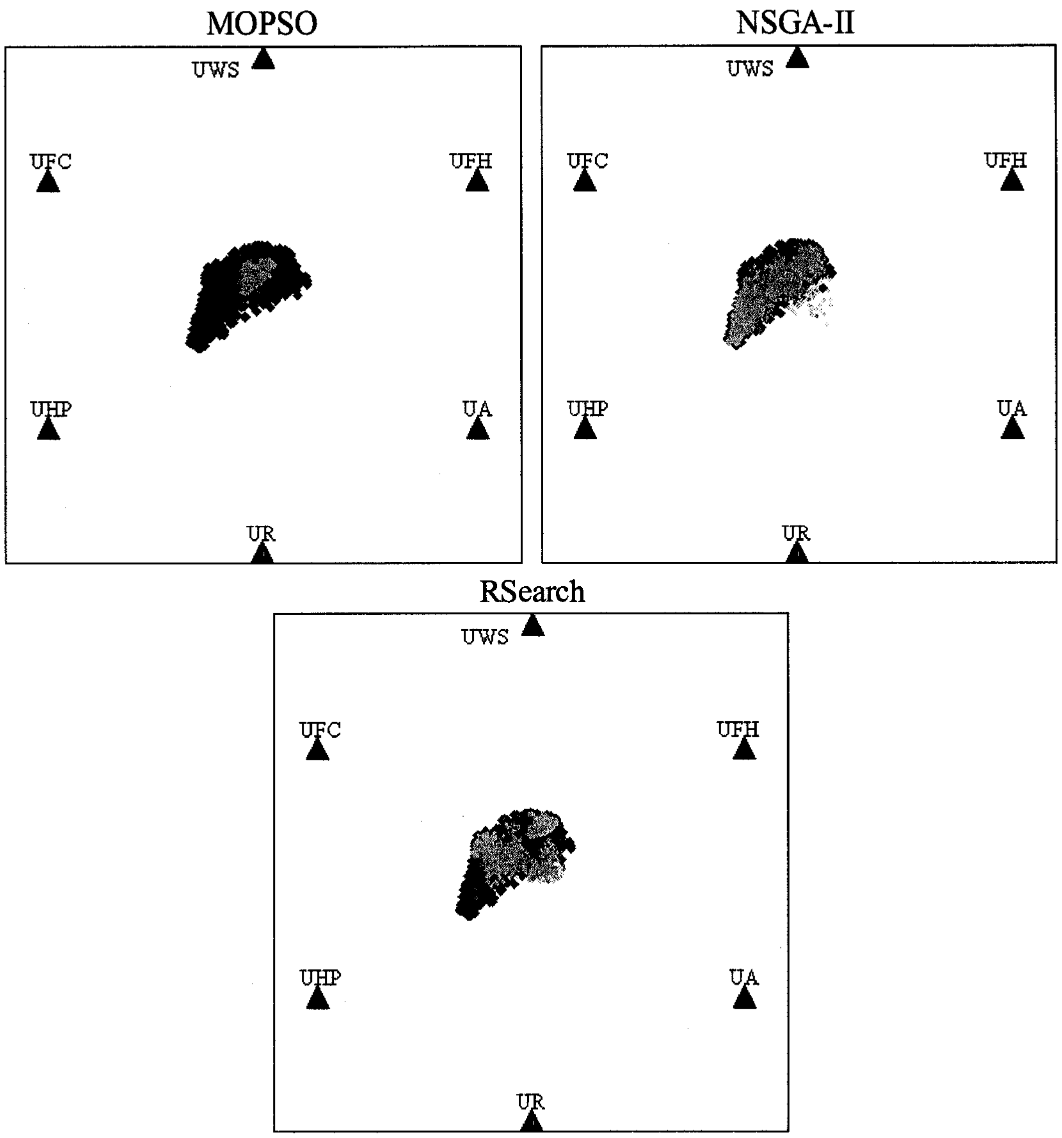



Figure 6.39 – Results for Test Problem FUZZY

Best Compromise Solutions

The best compromise solutions for seven different sets of weights are presented in Table 6.96, Figure 6.40 (all weights equal to one) and Figure 6.41 (best solutions for individual objectives). The associated guide curves are shown in Figure 6.42.

Table 6.96 – Best Compromise Solutions for Test Problem FUZZY

| Solution | UWS | UFC | UHP | UR | UA | UFH |
|---|---|---|---|---|---|---|
| (Comp) | 0.851 | 0.919 | 0.848 | 0.322 | 0.460 | 0.783 |
| (UWS) | 0.962 | 0.612 | 0.821 | 0.362 | 0.338 | 0.761 |
| (UFC) | 0.720 | 0.998 | 0.654 | 0.402 | 0.242 | 0.565 |
| (UHP) | 0.895 | 0.737 | 0.919 | 0.298 | 0.471 | 0.812 |
| (UR) | 0.367 | 0.975 | 0.341 | 0.584 | 0.071 | 0.174 |
| (UA) | 0.779 | 0.739 | 0.615 | 0.300 | 0.612 | 0.531 |
| (UFH) | 0.895 | 0.740 | 0.909 | 0.320 | 0.461 | 0.825 |

Solution named (Comp) in Table 6.96 corresponds to the best compromise solution when all weights are equal to one. Solution (UWS) is the best solution for the municipal and industrial water supply objective, giving a weight of one to the UWS membership function and zero to all the others. UFC, UHP, UR, UA, and UFH refer to flood control, hydropower, rafting, angling, and fish habitat objectives, repectively. These solutions are identified in the same way in Figures 6.40, 6.41, and 6.42.

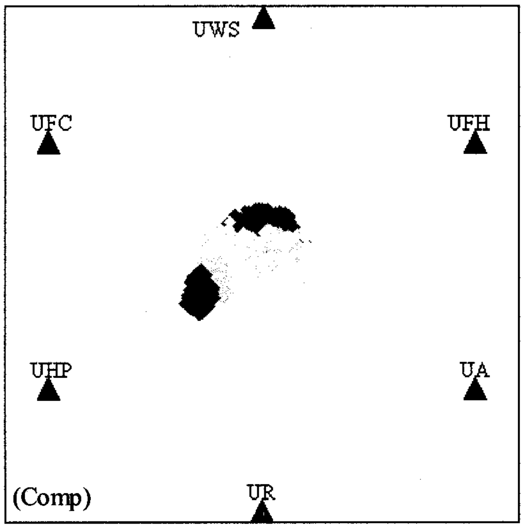



Figure 6.40 – Best Compromise Solution for Test Problem FUZZY

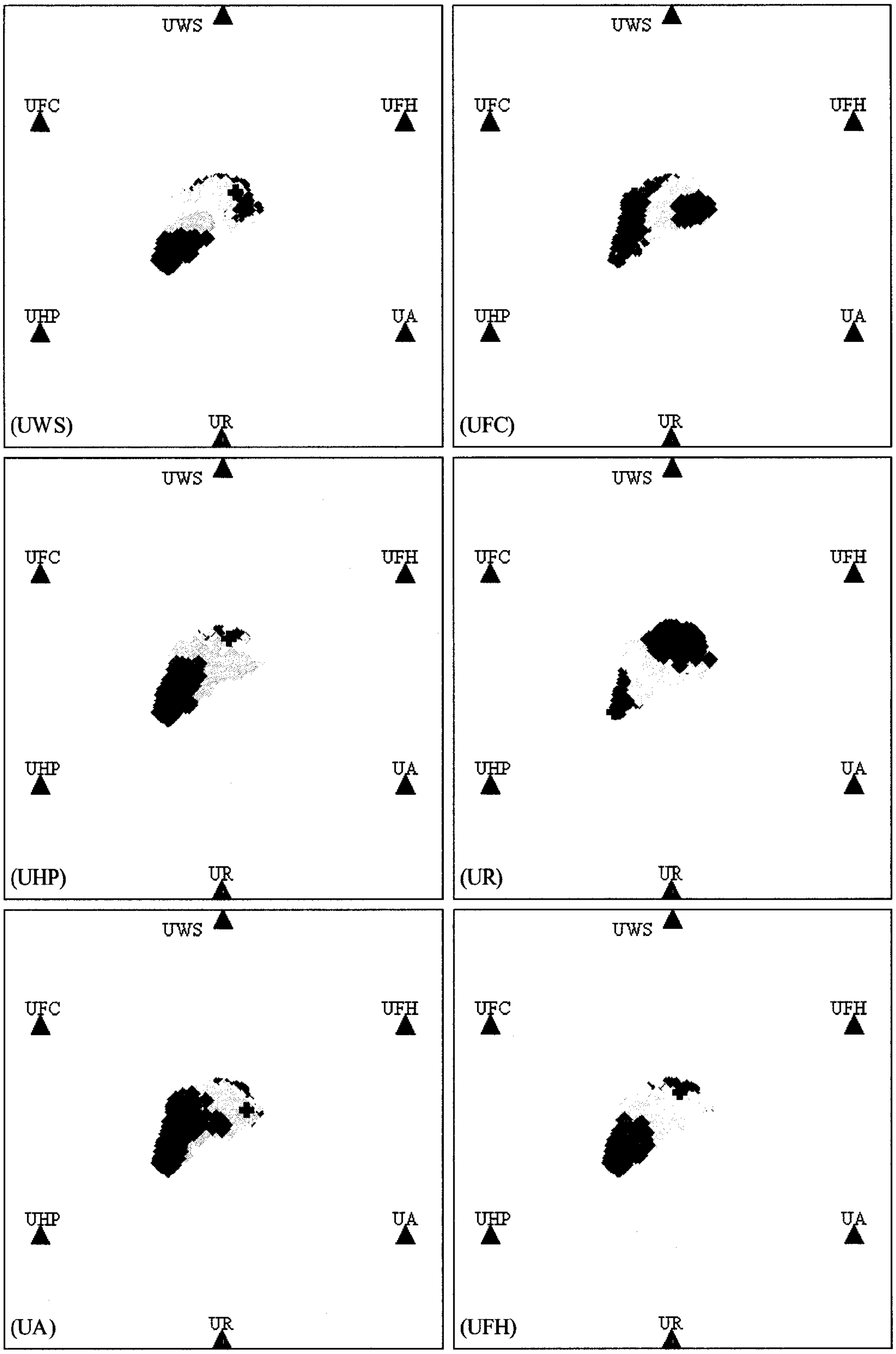



Figure 6.41 – Best Solutions for Each Objective for Test Problem FUZZY

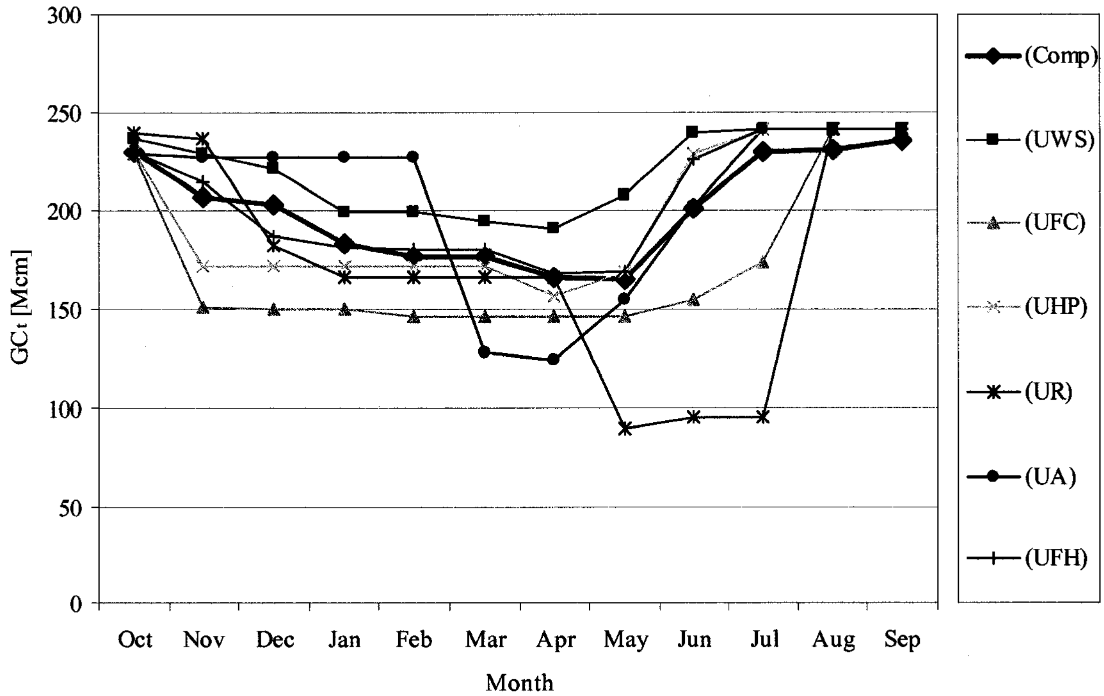



Figure 6.42 – Best Guide Curves for Test Problem FUZZY

6.2.4. Analysis of Results for Water Resources Management Problems

The selected water resources management problems present different levels of difficulty for multi-objective optimization. Overall, the evolutionary algorithms were able to find good approximations of the Pareto optimal sets, although some specific optimization issues were identified.

The NSGA-II seems to provide a more robust performance across the various types of problems than MOPSO. Two important remarks should be highlighted based on the results for the selected water resources management problems:

(i) For the test problems based on the RESIMWP model, the performance of the pure random search was quite close to the performance of the evolutionary algorithms. One possible explanation for this behavior is associated with the way the objective functions are defined for these problems, where all objectives are functions of only one of the decision variables, with the exception of the reliability objective which is not a direct function of any of the decision variables. All interactions or interferences of the decision variables on the performance of a specific solution occur completely through the constraints, conveying less information to the optimization algorithm than when the objective functions directly include all decision variables. This may be creating an effect similar to an optimization of a flat objective function, where no algorithm will do much better than a pure random search. Additional evidence of this behavior is the fact that pure random search also performed relatively well for Test Problem 5, which is the only one in the set of standard test functions that has the same characteristic in terms of the definition of the objective functions.

(ii) MOPSO and NSGA-II had problems dealing with the very uneven distribution of Pareto solutions imposed by the Test Problem TDT. The solutions from both evolutionary solvers presented strong clustering, although with very different patterns associated with the type of search mechanism of each algorithm. This is certainly an area where further research is needed.

Given that most multi-objective water resources problems can be mathematically formulated in many different ways, the potential applicability of MOPSO, or any other optimization method, to a specific problem has to be verified in a case-by-case manner. Some general guidelines could be applied, however:

(i) The evolutionary algorithms are definitely an advantageous alternative to mathematical programming (especially gradient-based search algorithms) when the problems involve discontinuities and high degree of non-linearity.

(ii) As with any other optimization technique, the performance of the evolutionary algorithms tend to worsen as the number of decision variables increase. Furthermore, the solvers developed in this research are not suitable for problems with very large number of decision variables, due to the computational efficiency of the spreadsheet platform and the non-compiled programming language (VBA). In this research, the largest number of decision variables ever tested was fourteen.

## 6.3. Sensitivity Analysis of the Parameters

Sensitivity analyses were performed to evaluate the response of both evolutionary solvers, MOPSO and NSGA-II, to changes on their main parameters. Test Problem 1, 5, TD and WEF were used for this purpose. Selected parameters were varied below and above the reference value used in the resolution of each problem.

For MOPSO, the following parameters were analyzed: (i) inertia coefficient $w$, (ii) search coefficients $c_1$ and $c_2$, and (iii) mutation rate, *MutRate*. The selected NSGA-II parameters were: (i) probability of crossover, $P_{cross}$, (ii) distribution index for crossover, $\eta_c$, (iii) probability of mutation, $P_{mut}$, and (iv) distribution index for mutation, $\eta_m$.

Table 6.97 shows the variations imposed to all parameters.

Table 6.97 – Variations of Parameters for Sensitivity Analysis

| Parameter | Variation 1 | Variation 2 | Variation 3 | Variation 4 | Reference |
|---|---|---|---|---|---|
| $W$ | 0.9-0.2 | 0.2 | 0.6 | 0.8 | 0.4 |
| $C_1 = c_2^*$ | 0.5 | 0.75 | 1.5 | 2.0 | 1.0 |
| *MutRate* | 0 | 0.25 | 0.75 | 1.0 | 0.5 |
| $P_{cross}$ | 0.25 | 0.5 | 0.75 | 1.0 | 0.9 |
| $\eta_c$ | 1 | 5 | 10 | 50 | 20 |
| $P_{mut}^*$ | 0 | 0.25 | 0.75 | 1.0 | 0.5 |
| $\eta_m$ | 1 | 5 | 10 | 50 | 20 |

* For Test Problem WEF, the variations of $c_1$&$c_2$ were 0.5, 1.0, 1.75, 2.0, with reference value of 1.4, and the variations for $P_{mut}$ were 0, 0.2, 0.75, 1.0, with reference value 0.333.

For the first variation of the MOPSO inertia coefficient, 0.9-0.2, the parameter was varied linearly from 0.9 to 0.2, as a function of the cycles.

The solvers were run thirty times for each problem, each variation of the parameters, keeping all other parameters at their reference values. The average values for all metrics were analyzed and parameterized graphs were developed, by dividing the average values of the metrics with changed parameters by the original average values of the metrics, i.e. all parameters at their reference values. All graphs have the same scale so graphical comparisons can be easily made. One of these graphs is shown in Figure 6.43, and all of the graphs are presented in Appendix B.

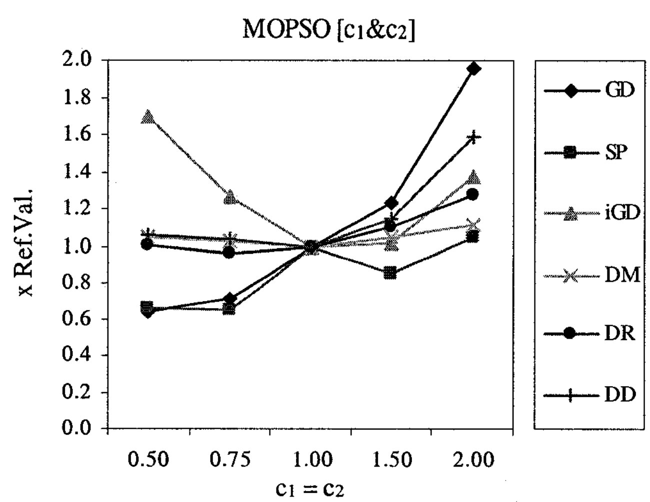



Figure 6.43 – Graph of Sensitivity Analysis for Test Problem 1

The following remarks can be made based on the results of the sensitivity analysis:

(i) The most sensitive parameters for MOPSO are the search coefficients $c_1$ and $c_2$, and they certainly have an important role balancing convergence and coverage; usually small values favor convergence (decreasing GD and DD) but worsen coverage (increasing iGD and DI).

(ii) The inertia coefficient is less sensitive and the mutation rate is by far the least sensitive parameter in MOPSO. In fact, this raises the question of whether the mutation operator is adequate and effective at all.

(iii) The NSGA-II Solver is more sensitive to its parameters than MOPSO. The probability of mutation seems to be the most sensitive parameter, while the distribution index for mutation is the least sensitive.

(iv) The default parameter values for both MOPSO and NSGA-II seem to provide an adequate compromise across all metrics, although some fine tuning can improve the performance for specific problems.

# 7. CONCLUSIONS AND RECOMMENDATIONS

Multi-objective particle swarm optimization is one of the most recent techniques within the family of evolutionary algorithms. Different MOPSO implementations have been reported in the EA-specialized literature with very few applications to real-world problems however.

In this research, a MOPSO implementation is proposed and applied to various EA standard test functions and to selected water resources management problems as well.

The application to the test functions had the primary objective of validating the developed codes by comparison with results of other MOPSO implementations reported in the literature, using similar parameters and performance metrics. The MOPSO was implemented as a generalized multi-objective solver, in the form of an add-in to Microsoft Excel®. This enhances the potential applicability of the tool. For instance, during the course of this research, after only two communications in international conferences with relatively restricted audiences, five people (two in the US, two in Europe, and one in Brazil) have requested copies of the MOPSO add-in.

Another three similar multi-objective solver add-ins to Excel were also developed. A second EA approach, the well known non-dominated sorting genetic algorithm II (NSGA-II), a traditional mathematical programming method, the ε-constraint with nonlinear optimization (ε-NLP), and a pure random search approach (RSearch). The NSGA-II is likely the most tested multi-objective evolutionary algorithm, with some practical applications reported in the water resources literature.

All these methods were applied to the set of selected test problems and their results were compared using different performance metrics. Each performance metric measures certain characteristics of the results, with specific advantages and limitations.

In most problems, the MOPSO and the NSGA-II algorithms provided good approximations to the true Pareto optimal sets. The ε-constraint method was used to generate the true Pareto optimal sets whenever possible, since this method uses nonlinear mathematical programming and necessary optimality conditions can be verified. However, the ε-constraint method is highly susceptible to the continuity and shape of the Pareto surfaces. In fact, for some of the selected water resources problems the ε-NLP approach presented very poor results. Overall, the NSGA-II algorithm seems to be more robust performing well in a wider range of problems, although MOPSO showed better performance for some of the problems.

The main advantage of the MOPSO is certainly associated with the simplicity of the algorithm. The basic MOPSO algorithm is much simpler and easier to implement than the NSGA-II. This makes MOPSO more flexible to accommodate necessary changes to deal with specific problems. For example, in Chapter 4 of this dissertation, a routine for dealing with equality constraints imposed upon decision variables was introduced. Without such a routine, most EA methods would simply fail to solve problems where those constraints are present, which is the case in many real-world applications. This routine was much easier to implement in MOPSO than NSGA-II, and the impact on the efficiency (processing time) of the method is also much smaller for MOPSO than NSGA-II. The use of dynamically defined bounds for decision variables, which allowed the solution of the multi-purpose reservoir operation problem based on guide curves and fuzzy objectives, is also significantly more difficult to implement for NSGA-II. That is because the crossover operator always works simultaneously with a pair of solutions to generate a new pair of solutions, and the bounds for each decision variable must be the same. The dynamic bound approach had to be implemented in NSGA-II with limitations, introducing a change in the crossover mechanism.

Both methods, MOPSO and NSGA-II, could not handle appropriately one of the water resources test problems where a very uneven distribution of solutions in the objective space existed. The responses of the two methods to that difficulty were completely different, however. While MOPSO was attracted to the region with higher concentration of solutions, with a strong clustering effect on that region, the NSGA-II departed from that region hindering its coverage of the Pareto optimal set.

Another line of investigation in this research was associated with the second phase of a multi-objective decision making process. After a reasonable number of Pareto optimal solutions are identified, ideally covering the whole extension of the feasible trade-offs, other methods have to be used to help the decision makers explore and select one or a subset among the several non-dominated solutions. For that purpose, the Interactive Compromise Coordinate (ICC) method was introduced.

The ICC method allows the visualization of all alternative solutions in a single unit circle graph. The decision maker can explore multi-objective Pareto optimal sets and rank alternatives using a compromise programming approach based on weights that can be interactively changed. The ICC method was included as part of all multi-objective solver add-ins developed in this research, and also as a stand-alone add-in to Excel that can be applied no matter what MCDA method was used to rank alternative solutions. The only assumptions are that the DM's preference structure can be modeled by a set of weights and that all alternatives are transitively comparable to each other, i.e. a complete pre-order is obtainable.

This combined package of EA algorithm and ICC method is the key contribution of this research. A much bigger problem is solved first by MOPSO or NSGA-II to characterize all possible trade-offs between objectives without setting any preference relation among them. Once this is done, the decision maker can easily use the ICC method to explore these trade-offs, interactively introduce relative preferences, and select one or a group of alternatives.

Finally, some specific recommendations are made for future research:

(i) Incorporate other diversity enhancing mechanisms for MOPSO that could be more efficient than the fitness sharing approach, for example, the adaptive grid used by Coello Coello et al. (2004) and Pulido (2005).

(ii) Study other approaches to improve MOPSO results in terms of coverage of the whole Pareto optimal set. Baumgartner et al. (2004) and Pulido (2005) investigated the use of sub-swarms for that purpose, with various degrees of success however.

(iii) Further investigate the MOPSO mutation operator. The sensitivity analysis conducted in this research using four different test problems showed that MOPSO results were very insensitive to the value of the mutation rate.

(iv) Study the performance of MOPSO and NSGA-II on other problems with very uneven distribution of solutions in the objective space, such as the selective withdrawal problem with temperature, dissolved oxygen, total dissolved solids objectives discussed here.

(v) Investigate the use of MOPSO and NSGA-II in other real-world problems with higher dimensionality in the decision space. The largest number of decision variables considered in this research was fourteen, for the multi-purpose reservoir operation test problem with fuzzy objectives.

(vi) The Grey Mountain Reservoir case study can be further explored by, for example, specifying the minimum values of the membership functions as objectives to be maximized instead of the average values. This max-min formulation could be appropriate on a more risk-averse decision environment. Also, the MOPSO performance on this problem might probably be improved through fine tuning of its parameters, especially the $c_1$ and $c_2$ search coefficients.

## REFERENCES


Alidi, A.S., and Al-Faraj, T.N. (1994). "Goal Programming Model for Distributing and Blending Desalinized Water." *J. Water Resour. Plng. and Mgmt.*, ASCE, Reston, VA, 120 (3) 267-275.

Atiquzzaman, M., Liong, S.-Y., and Yu, X. (2006). "Alternative Decision Making in Water Distribution Network with NSGA-II." *J. Water Resour. Plng. and Mgmt.*, ASCE, Reston, VA, 132 (2) 122-126.

Baltar, A.M., and Fontane, D.G. (2006). "Exploring Solutions of Multi-Objective Optimization and Multi-Criteria Decision Analysis: The Interactive Compromise Coordinate Method", submitted to the Journal of Multi-Criteria Decision Analisys in July 2006, under review.

Baumgartner, U., Magele, C., and Renhart, W. "Pareto Optimality and Particle Swarm Optimization." *IEEE Transactions on Magnetics*, IEEE, Piscataway, NJ, 40 (2) 1172-1175.

Benitez, J.E.A., Everson, R.M., and Fieldsend, J.E. (2005). "A MOPSO Algorithm Based Exclusively on Pareto Dominance Concepts." *Proc. of Evolutionary Multi-Criterion Optimization Third International Conf.*, Springer-Verlag, 459-473.

Brans, J.P., and Vincke, P.H. (1985). "A Preference Ranking Organization Method: The PROMETHEE Method for Multiple Criteria Decision-Making." *Management Science*, INFORMS, Evanston, IL, 31 (6) 647-656.Bohan, J.P. and Grace, J.L. (1973). "Selective Withdrawal from Man-Made Lakes". *Technical Report H-73-4*. U.S. Army Engineer Waterways Experiment Station, Vicksburg, MS.

Burn, D.H., and Yulianti, J.S. (2001). "Waste-Load Allocation Using Genetic Algorithms." *J. Water Resour. Plng. and Mgmt.*, ASCE, Reston, VA, 127 (2) 121-129.Can, E.K., and Houck, M.H. (1984). "Real-Time Reservoir Operations by Goal Programming." *J. Water Resour. Plng. and Mgmt.*, ASCE, Reston, VA, 110 (3) 297-309.

Carwell, H., Jager, H.I., and Sale, M.J. (1996). "Designing Instream Flows to Satisfy Fish and Human Water Needs." *J. Water Resour. Plng. and Mgmt.*, ASCE, Reston, VA, 122 (5) 356-363.

Cieniawski, S.E., Eheart, J.W., and Ranjithan, S. (1995). "Using Genetic Algorithms to Solve a Multiobjective Groundwater Monitoring Problem." *Water Resources Research*, AGU, Washington, DC, 31 (2) 399-409.

Cleveland, W.S. (1994). *The Elements of Graphing Data*. AT&T Bell Laboratories, Murray Hill, NJ, 297pp.

Coello Coello, C.A. (2001). "A Short Tutorial on Evolutionary Multiobjective Optimization." *Lect. Notes Comp. Sci. 1993*, Springer-Verlag, Berlin, Germany, 21-40.

Coello Coello, C.A.., and Lechuga, M.S. (2002). "MOPSO: A Proposal for Multiple Objective Particle Swarm Optimization." *Proc. Congress on Evolutionary Computation*, IEEE, Piscataway, NJ, Vol. 2, 1051-1056.

Coello Coello, C.A., Veldhuizen, D.A.V., and Lamont, G.B. (2002). *Evolutionary Algorithms for Solving Multi-Objective Problems*. Kluwer Academic Publishers, New York, NY, 576pp.

Coello Coello, C.A., Pulido, G.T., and Lechuga, M.S. (2004). "Handling Multiple Objectives with Particle Swarm Optimization." *IEEE Transactions on Evolutionary Computation*, IEEE, Piscataway, NJ, 8 (3) 256-279.

Cohon, J.L. and Marks, D.H. (1973). "Multiobjective Screening Models and Water Resource Investment." *Water Resources Research*, AGU, Washington, DC, 9 (4) 826-836.

Cohon, J.L. and Marks, D.H. (1975). "A Review and Evaluation of Multiobjective Programming Techniques." *Water Resources Research*, AGU, Washington, DC, 11 (2) 208-220.

Cordeiro Netto, O., Parent, E., and Duckstein, L. (1996). "Multicriterion Design of Long-Term Water Supply in Southern France." *J. Water Resour. Plng. and Mgmt.*, ASCE, Reston, VA, 122 (6) 403-413.

Deb, K. (1999). "Multi-objective Genetic Algorithms: Problem Difficulties and Construction of Test Problems". *Evolutionary Computation*, MIT Press, Cambridge, MA, 7 (3) 205-230.

Deb, K. (2001). *Multi-Objective Optimization using Evolutionary Algorithms*. J. Wiley & Sons, Chichester, England, 515pp.

Deb, K., and Agrawal, S. (1995). "Simulated Binary Crossover for Continuos Search Space." *Complex Systems*, Complex Systems Publications, Champaign, IL, 9 (2) 115-148.

Deb, K., Agrawal, S., Pratab, A., and Meyarivan, T. (2002). "A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II." *IEEE Transactions on Evolutionary Computation*, IEEE, Piscataway, NJ, 6 (2) 182-197.

Deb, K., Agrawal, S., Pratab, A., and Meyarivan, T. (2000). *A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II.* KanGAL Report No. 200001, Indian Institute of Technology, Kanpur, India, 11pp.

Dickson, G.W., Senn, J.A., and Chervany, N.L. (1977). "Research in Management Information Systems: The Minnesota Experiments." *Management Science*, INFORMS, Hanover, MD, 23 (9) 913-923.

Duckstein, L., Treichel, W., and Magnouni, S.E. (1994). "Ranking Ground-water Management Alternatives by Multicriterion Analysis." *J. Water Resour. Plng. and Mgmt.*, ASCE, Reston, VA, 120 (4) 546-565.

Elbeltagi, E., Hegazy, T., and Grierson, D. (2005). "Comparison Among Five Evolutionary-based Optimization Algorithms." *Adv. Engineering Informatics*, Elsevier, 19 (1) 43-53.

Elshorbagy, A. (2006). "Multicriterion Decision Analysis Approach to Assess the Utility of Watershed Modeling for Management Decisions." *Water Resources Research*, AGU, Washington, DC, 42 (9), 10.1029/2005WR004264.

Eschenbach, E.A., Magee, T., Zagona, E. Goranflo, M. and Shane, R. (2001). "Goal Programming Decision Support System for Multiobjective Operation of Reservoir Systems." *J. Water Resour. Plng. and Mgmt.*, ASCE, Reston, VA, 127 (2) 108-120.

Farmani, R., Walters, G.A., and Savic, D.A. (2005). "Trade-Off between Total Cost and Reliability for Anytown Water Distribution Network." *J. Water Resour. Plng. and Mgmt.*, ASCE, Reston, VA, 131 (3) 161-171.

Fieldsend, J.E., and Singh, S. (2002). "A Multi-Objective Algorithm based upon Particle Swarm Optimization, an Efficient Data Structure and Turbulence." *Proc. of 2002 U.K. Workshop on Computational Intelligence*, Birmingham, England, 37-44.

Fonseca, C.M. and Fleming, P.J. (1995). "An Overview of Evolutionary Algorithms in Multiobjective Optimization." *Evolut. Comptn.*, MIT Press, Cambridge, MA, 3 (1) 1-16.

Fontane, D.G. (2002). "Multi-objective Simulation and Optimization of Reservoir Operation Using Excel." *Proc. 2nd Joint Fed. Interag. Hydrologic Modlng. Conf.*, Las Vegas, NE.

Fontane, D.G., Gates, T.K., and Moncada, E. (1997). "Planning Reservoir Operations with Imprecise Objectives." *J. Water Resour. Plng. and Mgmt.*, ASCE, Reston, VA, 123 (3) 154-162.

Fontane, D.G. and Schneider, M.L. (1994). "Description of the Application of Goal Programming for Selective Withdrawal Structure Application." *Proc. 21st Annual ASCE Water Resources Plng. and Mgmt. Division Conf.* (Fontane, D.G. and Tuvel, H.N.,eds.), Denver, CO.

Geoffrion, A.M., Dyer, J.S., and Feinberg, A. (1972). "An Interactive Approach for Multi-Criterion Optimization, with an Application to the Operation of an Academic Department." *Management Science*, INFORMS, Hanover, MD, 19 (4) 357-368.

Georgakakos, A.P. (1993). "Operational Trade-Offs in Reservoir Control." *Water Resources Research*, AGU, Washington, DC, 29 (11) 3801-3819.

Gershon, M., and Duckstein, L. (1983). "Multiobjective Approaches to River Basin Planning." *J. Water Resour. Plng. and Mgmt.*, ASCE, Reston, VA, 109 (1) 13-28.

Gill, M.K., Kaheil, Y.H., Khalil, A., McKee, M., and Bastidas, L. (2006). "Multiobjective particle swarm optimization for parameter estimation in hydrology." *Water Resources Research*, AGU, Washington, DC, 42 (7), 10.1029/2005WR004528.

Goicoechea, A., Hansen, D.R., and Duckstein, L. (1982). *Multiobjective Decision Analysis with Engineering and Business Applications*. J. Wiley & Sons, New York, NY, 519pp.

Gupta, H.V., Sorooshian, S., and Yapo, P.O. (1998). "Toward Improved Calibration of Hydrologic Models: Multiple and Noncommensurable Measures of Information." *Water Resources Research*, AGU, Washington, DC, 34 (4) 751-763.

Halhal, D., Walters, G.A., Ouazar, D., and Savic, D.A. (1997). "Water Network Rehabilitation with Structured Messy Genetic Algorithm." *J. Water Resour. Plng. and Mgmt.*, ASCE, Reston, VA, 123 (3) 137-146.

Hu, X., Eberhart, R.C., and Shi, Y. (2003). "Particle Swarm with Extended Memory for Multiobjective Optimization." *Proc. of IEEE Swarm Intelligence Symposium*, IEEE, Piscataway, NJ, 193-197.

Huang, V.L., Suganthan, P.N., and Liang, L.L. (2006). "Comprehensive Learning Particle Swarm Optimizer for Solving Multiobjective Optimization Problems." *International Journal of Intelligent Systems*, Wiley Periodicals, Inc., Wilmington, DE, 21 (2) 209-226.

Hyde, K.M., Maier, H.R., and Colby, C.B. (2004). "Reliability-Based Approach to Multicriteria Decision Analysis for Water Resources." *J. Water Resour. Plng. and Mgmt.*, ASCE, Reston, VA, 130 (6) 429-438.

Jung, B.S. and Karney, B.W. (2006). "Hydraulic Optimization of Transient Protection Devices Using GA and PSO Approaches." *J. Water Resour. Plng. and Mgmt.*, ASCE, Reston, VA, 132 (1) 44-52.

Kasanen, E., Östermark, R., and Zeleny, M. (1991). "Gestalt System of Holistic Graphics: New Management Support View of MCDM." *Computers & Operations Research*, Elsevier, 18 (2) 233-239.

Kapelan, Z.S., Savic, D.A., and Walters, G.A. (2005). "Optimal Sampling Design Methodologies for Water Distribution Model Calibration." *J. Hydraulic Engrg.*, ASCE, Reston, VA, 131 (3) 190-200.

Kennedy, J. and Eberhart, R. (1995). "Particle Swarm Optimization." *Proc. 4$^{th}$ IEEE Int. Conf. on Neural Networks*, IEEE, Piscataway, NJ, 1942-1948.

Kita, H., Yabumoto, Y., Mori, N., Nishikawa, Y. (1996). "Multi-objective optimization by means of the thermodynamical genetic algorithm." Parallel Problem Solving from Nature-PPSN IV, H.-M. Voigt et al., Eds., Springer, New York, NY, 504-512.

Ko, S.-K., Fontane, D.G., and Labadie, J.W. (1992). "Multiobjective Optimization of Reservoir Systems Operation." *Water Resources Bulletin*, AWRA, Bethesda, MD, 28 (1) 111-127.

Ko, S.-K., Oh, M.H., and Fontane, D.G. (1997). "Multiobjective Analysis of Service-Water-Transmission Systems." *J. Water Resour. Plng. and Mgmt.*, ASCE, Reston, VA, 123 (2) 78-83.

Kursawe, F. (1991). "A Variant of Evolution Strategies for Vector Optimization." *Lect. Notes Comp. Sci. 1991*, Springer-Verlag, Berlin, Germany, 193-197.

Kwanyuen, B., and Fontane, D.G. (1998). "Heuristic Branch-and-Bound Method for Ground Water Development Planning." *J. Water Resour. Plng. and Mgmt.*, ASCE, Reston, VA, 124 (3) 140-148.

Laabs, H. and Schultz, G.A. (1992). "Reservoir Management Rules Derived with the Aid of Multiple Objective Decision Making Techniques." *Water Resources Bulletin*, AWRA, Bethesda, MD, 28 (1) 211-222.

Lechuga, M.S., and Rowe, J.E. (2005). "Particle Swarm Optimization and Fitness Sharing to Solve Multi-Objective Optimization Problems." *Proc. of IEEE Congress on Evolutionary Computation CEC'05*, IEEE, Piscataway, NJ, 1204-1211.

Li, X. (2003). "A Non-dominated Sorting Particle Swarm Optimizer for Multiobjective Optimization." *Proc. of the Genetic and Evolutionary Computation Conf. GECCO 2003*, Springer-Verlag, 37-48.

Liong, S., Khu, S., and Chan, W. (2001). "Derivation of Pareto Front with Genetic Algorithm and Neural Network." *J. Hydrologic Engrg.*, ASCE, Reston, VA, 6 (1) 52-61.

Loganathan, G.V., and Bhattacharya, D. (1990). "Goal-Programming Techniques for Optimal Reservoir Operations." *J. Water Resour. Plng. and Mgmt.*, ASCE, Reston, VA, 116 (6) 820-838.

Losa, F.B., Van Den Honert, R., and Joubert, A. (2001). "The Multivariate Analysis Biplot as a Tool for Conflict Analysis in MCDA." *Journal of Multi-Criteria Decision Analysis*, John Wiley & Sons, Hoboken, NJ, 10 (5) 273-284.

Louie, P.W.F., Yeh, W.W.-G., and Hsu, N.-S. (1984). "Multiobjective Water Resources Management Planning." *J. Water Resour. Plng. and Mgmt.*, ASCE, Reston, VA, 110 (1) 39-56.

Mao, N., and Mays, L.W. (1994). "Goal Programming Models for Determining Freshwater Inflows to Estuaries." *J. Water Resour. Plng. and Mgmt.*, ASCE, Reston, VA, 120 (3) 316-329.

Major, D.C. (1969). "Benefit-Cost Ratios for Projects in Multiple Objective Investment Programs." *Water Resources Research*, AGU, Washington, DC, 5 (6) 1175-1178.

Mason, R.O., and Mitroff, I.I. (1973). "A Program for Research on Management Information Systems." *Management Science*, INFORMS, Hanover, MD, 19 (5) 475-487.

McMahon, T.A. and Mein, R.G. (1988). *River and Reservoir Yield.* Water Resources Publications, Littleton, CO, 368 pp.

McPhee, J., and Yeh, W.W.-G. (2004). "Multiobjective Optimization for Sustainable Groundwater Management in Semiarid Regions." *J. Water Resour. Plng. and Mgmt.*, ASCE, Reston, VA, 130 (6) 490-497.

Mendes, R., Kennedy, J., and Neves, J. (2004). "The Fully Informed Particle Swarm: Simpler, Maybe Better." *IEEE Transactions on Evolutionary Computation*, IEEE, Piscataway, NJ, 8 (3) 204-210.

Microsoft. (2003). Microsoft® Office Excel 2003. Microsoft Corp., Redmond, WA.

Miettinen, K. (2001). "Some Methods for Nonlinear Multi-objective Optimization." Springer-Verlag, *Lecture Notes in Computer Science 1993*, Berlin, Germany, 1-20.

Moncada, E. (1999). *Planning Reservoir Operations with Imprecise Objectives*. Ph.D. Dissertation, Department of Civil Engineering, Colorado State University, Fort Collins, CO, 199pp.

Moore, J., and Chapman, R. (1999). *Application of Particle Swarm to Multiobjective Optimization*. Department of Computer Science and Software Engineering, Auburn University, Auburn, AL, 4pp.

Mostaghim, S., and Teich, J. (2003). "Strategies for Finding Good Local Guides in Multi-objective Particle Swarm Optimization (MOPSO)." *Proc. of IEEE Swarm Intelligence Symposium*, IEEE, Piscataway, NJ, 26-33.

Moy, W.-S., Cohon, J.L., and ReVelle, C.S. (1986). "A Programming Model for Analysis of the Reliability, Resilience, and Vulnerability of a Water Supply Reservoir." *Water Resources Research*, AGU, Washington, DC, 22 (4) 489-498.

Muleta, M.K. and Nicklow, J.W. (2005). "Decision Support for Watershed Management Using Evolutionary Algorithms." *J. Water Resour. Plng. and Mgmt.*, ASCE, Reston, VA, 131 (1) 35-44.

Olsen, J.R., Beling, P.A., and Lambert, J.H. (2000). "Dynamic Models for Floodplain Management." *J. Water Resour. Plng. and Mgmt.*, ASCE, Reston, VA, 126 (3) 167-175.

Parsopoulos, K.E., and Vrahatis, M.N. (2002). "Particle Swarm Optimization Method in Multiobjective Problems." *Proc. ACM Symposium on Applied Computing*, ACM, New York, NY, 603-607.

Prasad, T.D., and Park, N.-S. (2004). "Multiobjective Genetic Algorithms for Design of Water Distribution Networks." *J. Water Resour. Plng. and Mgmt.*, ASCE, Reston, VA, 130 (1) 73-82.

Prasad, T.D., Walters, G.A., and Savic, D.A. (2004). "Booster Disinfection of Water Supply Networks: Multiobjective Approach." *J. Water Resour. Plng. and Mgmt.*, ASCE, Reston, VA, 130 (5) 367-376.

Pulido, G.T. (2005). *On the Use of Self-Adaptation and Elitism for Multiobjective Particle Swarm Optimization*. Ph.D. Dissertation, Electrical Engineering Department, Centro de Investigación y Estudios Avanzados del Instituto Politécnico Nacional, Mexico City, Mexico, 259pp.

Pulido, G.T., and Coello Coello, C.A. (2004). "Using Clustering Techniques to Improve the Performance of a Multi-Objective Particle Swarm Optimizer." *Proc. Of the Genetic and Evolutionary Computation Conf. GECCO 2004*, Springer-Verlag, 225-237.

Raju, K.S., Duckstein, L., and Arondel, C. (2000). "Multicriterion Analysis for Sustainable Water Resources Planning: A Case Study in Spain." *Water Resources Management*, Kluwer Academic Publishers, Netherlands, 14 (6) 435-456.

Ray, T., and Liew, K.M. (2002). "A Swarm Metaphor for Multiobjective Design Optimization." *Engineering Optimization*, Taylor & Francis Ltd, Abingdon, England, 34 (2) 141-153.

Ritzel, B.J., Eheart, J.W., and Ranjithan, S. (1994). "Using Genetic Algorithms to Solve a Multiple Objective Groundwater Pollution Containment Problem." *Water Resources Research*, AGU, Washington, DC, 30 (5) 1589-1603.

Roy, B., Slowinski, R., and Treichel, W. (1992). "Multicriteria Programming of Water Supply Systems for Rural Areas." *Water Resources Bulletin*, AWRA, Bethesda, MD, 28 (1) 13-31.

Schott, J.R. (1995). *Fault Tolerant Design Using Single and Multicriteria Genetic Algorithm Optimization.* M.S. Thesis, MIT, Cambridge, MA.

Shafike, N.G., Duckstein, L., and Maddock III, T. (1992). "Multicriterion Analysis of Groundwater Contamination Management." *Water Resources Bulletin*, AWRA, Bethesda, MD, 28 (1) 33-43.

Simon, H.A. (1957). *Administrative Behavior: A Study of Decision-Making Processes in Administrative Organization.* The Macmillan Company, New York, NY, 259pp.

Sobol, M.G., and Klein, G. (1989). "New Graphics as Computerized Displays for Human Information Processing." *IEEE Transactions on Systems, Man, and Cybernetics*, IEEE, Piscataway, NJ, 19 (4) 893-898.

Srinivas, N., and Deb, K. (1995). "Multiobjective Optimization Using Nondominated Sorting in Genetic Algorithms." *Evolutionary Computation*, MIT Press, Cambridge, MA, 2 (3) 221-248.

Suen, J.-P., Eheart, J.W., and Herricks, E.E. (2005). "Integrating Ecological Flow Regimes in Water Resources Management Using Multiobjective Analysis." *Proc. ASCE/EWRI World Water and Envmtl. Resour. Cong.*, ASCE, Reston, VA.

Tan, Y.S., and Fraser, N.M. (1998). "The Modified Star Graph and the Petal Diagram: Two New Visual Aids for Discrete Alternative Multicriteria Decision Making." *Journal of Multi-Criteria Decision Analysis*, John Wiley & Sons, Hoboken, NJ, 7 (1) 20-33.

Tanaka, M. (1995). "GA-Based Decision Support System for Multicriteria Optimization." *Proc. of IEEE Int. Conf. Systems, Man and Cybernetics-2*, IEEE, Piscataway, NJ, 1556-1561.

Tang, Y. and Reed, P.M. (2005). "Multiobjective Tools and Strategies for Calibrating Integrated Models." *Proc. ASCE/EWRI World Water and Envmtl. Resour. Cong.*, ASCE, Reston, VA.

Tecle, A., Fogel, M., and Duckstein, L. (1988). "Multicriterion Selection of Wastewater Management Alternatives." *J. Water Resour. Plng. and Mgmt.*, ASCE, Reston, VA, 114 (4) 383-398.

Uspensky, J.V. (1937). Introduction to Mathematical Probability. McGraw-Hill, New York, NY, 411pp.

Van Veldhuizen, D.A. (1999). Multiobjective Evolutionary Algorithms: Classifications, Analices, and New Innovations." Ph.D. Dissertation, Dept. of Electrical and Computer Engr., Air Force Institute of Technology, Wright-Patterson AFB, OH.

Van Veldhuizen, D.A. and Lamont, G.B. (1998). "Multiobjective evolutionary algorithm research: A history and analysis." Tech. Rep. TR-98-03, Air Force Inst. Of Technol., Wright-Patterson AFB, OH.

Villalobos-Arias, M.A., Pulido, G.T., and Coello Coello, C.A. (2005). "A Proposal to Use Stripes to Maintain Diversity in a Multi-Objective Particle Swarm Optimizer." *Proc. of IEEE Swarm Intelligence Symposium SIS05*, IEEE, Piscataway, NJ, 22-29.

Wen, C.-G., and Lee, C.-S. (1998). "A Neural Network Approach to Multiobjective Optimization for Water Quality Management in a River Basin." *Water Resources Research*, AGU, Washington, DC, 34 (3) 427-436.

Yandamuri, S.R.M., Srinivasan, K., and Bhallamudi, S.M. (2006). "Multiobjective Optimal Waste Load Allocation Models for Rivers Using Nondominated Sorting Genetic Algorithm-II." *J. Water Resour. Plng. and Mgmt.*, ASCE, Reston, VA, 132 (3) 133-143.

Yeh, C.-H., and Labadie, J.W. (1997). "Multiobjective Watershed-Level Planning of Storm Water Detention Systems." *J. Water Resour. Plng. and Mgmt.*, ASCE, Reston, VA, 123 (6) 336-343.

Yeh, W.W-G. and Becker, L. (1982). "Multiobjective Analysis of Multireservoir Operations." *Water Resources Research*, AGU, Washington, DC, 18 (5) 1326-1336.

Zadeh, L.A. (1963). "Optimality and Non-Scalar-Valued Performance Criteria." *IEEE Transactions on Automatic Control*, IEEE, Piscataway, NJ, 8 (1) 59-60.

Zitzler, E., Deb, K., and Thiele, L. (2000). "Comparison of Multiobjective Evolutionary Algorithms: Empirical Results." *Evolutionary Computation*, MIT Press, Cambridge, MA, 8 (2) 173-195.

Zitzler, E. and Thiele, L. (1999). "Multiobjective Evolutionary Algorithms: A Comparative Case Study and the Strength Pareto Approach." *IEEE Transactions on Evolutionary Computation*, IEEE, Piscataway, NJ, 3 (4) 257-271.

# APPENDIX A: STATISTICS OF PERFORMANCE METRICS

Table A.1 – MOPSO Statistics for Test Problem 1

| Run | GD | SP | iGD | DI | PT | DR | DD |
|---|---|---|---|---|---|---|---|
| 1 | 0.007830 | 0.097528 | 0.005689 | 0.774303 | 5.95 | 0.51 | 0.70% |
| 2 | 0.013721 | 0.070532 | 0.005141 | 0.725994 | 6.08 | 0.45 | 0.83% |
| 3 | 0.003276 | 0.047236 | 0.006092 | 0.653412 | 6.48 | 0.67 | 0.57% |
| 4 | 0.003541 | 0.058344 | 0.005639 | 0.671965 | 6.47 | 0.48 | 0.70% |
| 5 | 0.006721 | 0.072195 | 0.005446 | 0.676204 | 6.33 | 0.61 | 0.68% |
| 6 | 0.012792 | 0.069946 | 0.004829 | 0.699801 | 5.56 | 0.59 | 0.75% |
| 7 | 0.003246 | 0.047414 | 0.006142 | 0.682197 | 5.88 | 0.51 | 0.62% |
| 8 | 0.006350 | 0.063797 | 0.005536 | 0.721324 | 6.61 | 0.64 | 0.61% |
| 9 | 0.007867 | 0.080037 | 0.005665 | 0.685721 | 6.70 | 0.54 | 0.70% |
| 10 | 0.025776 | 0.248502 | 0.005491 | 0.824343 | 5.88 | 0.62 | 0.69% |
| 11 | 0.003166 | 0.065924 | 0.005800 | 0.666495 | 5.94 | 0.53 | 0.57% |
| 12 | 0.003602 | 0.051055 | 0.004961 | 0.624752 | 5.70 | 0.58 | 0.54% |
| 13 | 0.003435 | 0.040680 | 0.005740 | 0.677925 | 6.02 | 0.56 | 0.50% |
| 14 | 0.033110 | 0.130470 | 0.005676 | 0.784664 | 5.75 | 0.56 | 1.05% |
| 15 | 0.081840 | 0.671279 | 0.005044 | 0.982214 | 5.20 | 0.69 | 1.27% |
| 16 | 0.018884 | 0.156041 | 0.005990 | 0.787215 | 6.06 | 0.48 | 0.83% |
| 17 | 0.003390 | 0.049227 | 0.005657 | 0.646571 | 5.61 | 0.49 | 0.62% |
| 18 | 0.006327 | 0.054605 | 0.005878 | 0.699064 | 6.12 | 0.70 | 0.62% |
| 19 | 0.137047 | 1.370109 | 0.005713 | 1.132237 | 4.56 | 0.58 | 1.88% |
| 20 | 0.004147 | 0.047110 | 0.005157 | 0.597727 | 5.75 | 0.61 | 0.58% |
| 21 | 0.003187 | 0.048089 | 0.005197 | 0.644602 | 6.00 | 0.54 | 0.47% |
| 22 | 0.004560 | 0.051212 | 0.005707 | 0.651074 | 6.20 | 0.55 | 0.59% |
| 23 | 0.005682 | 0.075729 | 0.005776 | 0.715662 | 6.36 | 0.59 | 0.60% |
| 24 | 0.003143 | 0.059154 | 0.005314 | 0.631007 | 8.02 | 0.57 | 0.69% |
| 25 | 0.009974 | 0.085565 | 0.005649 | 0.723240 | 6.66 | 0.58 | 0.66% |
| 26 | 0.006307 | 0.050374 | 0.005197 | 0.619600 | 6.19 | 0.74 | 0.74% |
| 27 | 0.004417 | 0.070663 | 0.005499 | 0.738754 | 5.98 | 0.63 | 0.54% |
| 28 | 0.125834 | 1.202198 | 0.006434 | 1.103371 | 4.94 | 0.58 | 1.69% |
| 29 | 0.003457 | 0.054847 | 0.005631 | 0.624244 | 6.58 | 0.55 | 0.60% |
| 30 | 0.003290 | 0.044039 | 0.004809 | 0.595958 | 6.50 | 0.58 | 0.59% |
| | | | | | | | |
| Best | | | | | | | |
| Worst | 0.003143 | 0.040680 | 0.004809 | 0.595958 | 4.56 | 0.45 | 0.47% |
| Average | 0.137047 | 1.370109 | 0.006434 | 1.132237 | 8.02 | 0.74 | 1.88% |
| Median | 0.018531 | 0.174463 | 0.005550 | 0.725388 | 6.07 | 0.58 | 0.75% |
| Std.Dev. | 0.005995 | 0.064860 | 0.005644 | 0.683959 | 6.04 | 0.58 | 0.64% |

Table A.2 – MOPSO-e Statistics for Test Problem 1

| Run | GD | SP | iGD | DI | PT | DR | DD |
|---|---|---|---|---|---|---|---|
| 1 | 0.003419 | 0.059469 | 0.005508 | 0.690361 | 8.48 | 0.51 | 0.65% |
| 2 | 0.013064 | 0.115074 | 0.005861 | 0.759930 | 7.77 | 0.53 | 0.63% |
| 3 | 0.003887 | 0.050732 | 0.005882 | 0.710546 | 8.27 | 0.57 | 0.53% |
| 4 | 0.003523 | 0.053649 | 0.004836 | 0.640046 | 8.14 | 0.66 | 0.70% |
| 5 | 0.003337 | 0.050095 | 0.005922 | 0.692848 | 9.36 | 0.62 | 0.52% |
| 6 | 0.003700 | 0.049443 | 0.006041 | 0.700847 | 9.39 | 0.61 | 0.65% |
| 7 | 0.003844 | 0.055238 | 0.005432 | 0.666757 | 8.36 | 0.61 | 0.60% |
| 8 | 0.051159 | 0.507462 | 0.005297 | 0.940727 | 7.89 | 0.59 | 0.98% |
| 9 | 0.007382 | 0.039822 | 0.011007 | 0.724463 | 8.27 | 0.49 | 0.53% |
| 10 | 0.008843 | 0.090543 | 0.005193 | 0.674676 | 9.02 | 0.52 | 0.63% |
| 11 | 0.003550 | 0.054058 | 0.005428 | 0.652958 | 8.12 | 0.52 | 0.44% |
| 12 | 0.058209 | 0.319714 | 0.005373 | 0.890745 | 7.44 | 0.52 | 1.57% |
| 13 | 0.149858 | 1.498565 | 0.004976 | 1.132643 | 7.12 | 0.59 | 1.90% |
| 14 | 0.012812 | 0.098104 | 0.005010 | 0.716805 | 7.67 | 0.65 | 0.68% |
| 15 | 0.009771 | 0.099370 | 0.005148 | 0.695976 | 8.81 | 0.56 | 0.67% |
| 16 | 0.003337 | 0.046638 | 0.005801 | 0.675964 | 8.33 | 0.51 | 0.65% |
| 17 | 0.034817 | 0.349223 | 0.005269 | 0.880279 | 7.64 | 0.55 | 0.84% |
| 18 | 0.003591 | 0.042201 | 0.005800 | 0.612450 | 8.70 | 0.59 | 0.52% |
| 19 | 0.008003 | 0.045275 | 0.006305 | 0.740251 | 8.25 | 0.62 | 0.73% |
| 20 | 0.006981 | 0.055881 | 0.006081 | 0.729305 | 8.70 | 0.59 | 0.57% |
| 21 | 0.003845 | 0.064941 | 0.006269 | 0.737263 | 8.38 | 0.53 | 0.54% |
| 22 | 0.004468 | 0.055966 | 0.005424 | 0.666364 | 8.34 | 0.53 | 0.53% |
| 23 | 0.003346 | 0.042356 | 0.005538 | 0.672072 | 8.16 | 0.68 | 0.46% |
| 24 | 0.005581 | 0.060932 | 0.006089 | 0.740886 | 8.17 | 0.53 | 0.57% |
| 25 | 0.011459 | 0.049529 | 0.006503 | 0.730747 | 8.86 | 0.63 | 0.81% |
| 26 | 0.082875 | 0.057091 | 0.005176 | 0.895133 | 7.47 | 0.63 | 1.63% |
| 27 | 0.006166 | 0.067359 | 0.005474 | 0.728092 | 8.58 | 0.48 | 0.62% |
| 28 | 0.005252 | 0.068558 | 0.005267 | 0.667373 | 8.44 | 0.46 | 0.50% |
| 29 | 0.003599 | 0.054262 | 0.006038 | 0.661659 | 9.02 | 0.61 | 0.66% |
| 30 | 0.004034 | 0.053757 | 0.005411 | 0.648511 | 8.23 | 0.59 | 0.62% |
| Best | 0.003337 | 0.039822 | 0.004836 | 0.612450 | 7.12 | 0.46 | 0.44% |
| Worst | 0.149858 | 1.498565 | 0.011007 | 1.132643 | 9.39 | 0.68 | 1.90% |
| Average | 0.017457 | 0.141844 | 0.005779 | 0.735889 | 8.31 | 0.57 | 0.73% |
| Median | 0.005416 | 0.055924 | 0.005491 | 0.705697 | 8.30 | 0.58 | 0.63% |
| Std.Dev. | 0.031346 | 0.277241 | 0.001077 | 0.109601 | 0.54 | 0.06 | 0.35% |

Table A.3 – MOPSO-p Statistics for Test Problem 1

| Run | GD | SP | iGD | DI | PT | DR | DD |
|---|---|---|---|---|---|---|---|
| 1 | 0.006846 | 0.044589 | 0.004570 | 0.561810 | 6.39 | 0.29 | 0.42% |
| 2 | 0.008283 | 0.063780 | 0.005205 | 0.708015 | 6.36 | 0.20 | 0.53% |
| 3 | 0.024318 | 0.260408 | 0.005745 | 0.777860 | 5.62 | 0.31 | 0.64% |
| 4 | 0.003211 | 0.049161 | 0.004573 | 0.627279 | 6.08 | 0.28 | 0.28% |
| 5 | 0.115966 | 1.062328 | 0.004789 | 1.060438 | 5.59 | 0.28 | 2.60% |
| 6 | 0.004153 | 0.077793 | 0.006018 | 0.672261 | 6.64 | 0.23 | 0.32% |
| 7 | 0.014421 | 0.140082 | 0.005175 | 0.716142 | 6.31 | 0.42 | 0.44% |
| 8 | 0.013381 | 0.157632 | 0.005269 | 0.696173 | 6.58 | 0.32 | 0.45% |
| 9 | 0.009691 | 0.063619 | 0.005950 | 0.657066 | 6.50 | 0.36 | 0.44% |
| 10 | 0.003275 | 0.066967 | 0.005095 | 0.651780 | 6.75 | 0.31 | 0.26% |
| 11 | 0.029040 | 0.194699 | 0.005454 | 0.813184 | 6.45 | 0.31 | 0.90% |
| 12 | 0.011594 | 0.117625 | 0.005898 | 0.727179 | 6.52 | 0.35 | 0.42% |
| 13 | 0.014474 | 0.121520 | 0.005608 | 0.764632 | 6.89 | 0.30 | 0.49% |
| 14 | 0.002949 | 0.042139 | 0.004900 | 0.609443 | 6.38 | 0.27 | 0.28% |
| 15 | 0.025660 | 0.093293 | 0.005164 | 0.732234 | 6.81 | 0.29 | 0.87% |
| 16 | 0.034849 | 0.151296 | 0.005141 | 0.784026 | 6.52 | 0.29 | 1.18% |
| 17 | 0.004867 | 0.062322 | 0.005021 | 0.671062 | 6.44 | 0.26 | 0.32% |
| 18 | 0.003552 | 0.054860 | 0.005346 | 0.718919 | 6.45 | 0.26 | 0.28% |
| 19 | 0.011077 | 0.050742 | 0.005532 | 0.701532 | 7.41 | 0.36 | 0.52% |
| 20 | 0.003729 | 0.045129 | 0.006730 | 0.626842 | 6.00 | 0.23 | 0.31% |
| 21 | 0.029753 | 0.232431 | 0.005286 | 0.825061 | 6.42 | 0.25 | 0.97% |
| 22 | 0.003180 | 0.061534 | 0.005472 | 0.715028 | 7.06 | 0.32 | 0.30% |
| 23 | 0.019172 | 0.115808 | 0.005876 | 0.786593 | 6.61 | 0.33 | 0.63% |
| 24 | 0.012983 | 0.092921 | 0.005716 | 0.676214 | 6.97 | 0.41 | 0.49% |
| 25 | 0.067972 | 0.522295 | 0.005137 | 0.953509 | 6.58 | 0.31 | 1.65% |
| 26 | 0.003773 | 0.068589 | 0.005818 | 0.663473 | 6.19 | 0.28 | 0.28% |
| 27 | 0.007521 | 0.048616 | 0.005690 | 0.673596 | 6.89 | 0.28 | 0.44% |
| 28 | 0.013528 | 0.056945 | 0.005013 | 0.678666 | 7.11 | 0.34 | 0.64% |
| 29 | 0.024836 | 0.095495 | 0.005500 | 0.758882 | 7.00 | 0.40 | 0.86% |
| 30 | 0.008738 | 0.123478 | 0.006724 | 0.742926 | 7.00 | 0.35 | 0.36% |
| | | | | | | | |
| Best | 0.002949 | 0.042139 | 0.004570 | 0.561810 | 5.59 | 0.20 | 0.002949 |
| Worst | 0.115966 | 1.062328 | 0.006730 | 1.060438 | 7.41 | 0.42 | 0.115966 |
| Average | 0.017893 | 0.144603 | 0.005447 | 0.725061 | 6.55 | 0.31 | 0.017893 |
| Median | 0.011336 | 0.085357 | 0.005400 | 0.711521 | 6.52 | 0.31 | 0.011336 |
| Std.Dev. | 0.022977 | 0.197858 | 0.000521 | 0.098850 | 0.41 | 0.05 | 0.022977 |

Table A.4 – NSGA-II Statistics for Test Problem 1

| Run | GD | SP | iGD | DI | PT | DR | DD |
|---|---|---|---|---|---|---|---|
| 1 | 0.013958 | 0.045052 | 0.006750 | 0.726109 | 5.81 | 0.49 | 0.79% |
| 2 | 0.029092 | 0.261862 | 0.011211 | 0.934712 | 5.73 | 0.60 | 1.04% |
| 3 | 0.010994 | 0.049261 | 0.007665 | 0.771956 | 5.75 | 0.59 | 0.71% |
| 4 | 0.016005 | 0.146149 | 0.007228 | 0.773769 | 5.73 | 0.62 | 0.77% |
| 5 | 0.056139 | 0.534565 | 0.006346 | 0.953072 | 5.73 | 0.55 | 1.27% |
| 6 | 0.054141 | 0.481631 | 0.011064 | 0.969811 | 5.75 | 0.56 | 1.20% |
| 7 | 0.159072 | 0.050490 | 0.010726 | 1.111619 | 5.73 | 0.59 | 2.77% |
| 8 | 0.003345 | 0.041924 | 0.006664 | 0.632339 | 5.72 | 0.58 | 0.60% |
| 9 | 0.055296 | 0.512529 | 0.006561 | 0.938096 | 5.72 | 0.59 | 1.22% |
| 10 | 0.003291 | 0.040702 | 0.006180 | 0.670657 | 5.78 | 0.57 | 0.61% |
| 11 | 0.026200 | 0.237068 | 0.005507 | 0.818129 | 5.77 | 0.49 | 0.97% |
| 12 | 0.010442 | 0.041707 | 0.012773 | 0.744924 | 5.73 | 0.56 | 0.65% |
| 13 | 0.036991 | 0.041090 | 0.008965 | 0.783257 | 5.80 | 0.59 | 1.36% |
| 14 | 0.013334 | 0.058410 | 0.006706 | 0.678000 | 5.83 | 0.63 | 0.86% |
| 15 | 0.036649 | 0.039352 | 0.009574 | 0.854237 | 5.78 | 0.48 | 1.24% |
| 16 | 0.071835 | 0.219469 | 0.005673 | 0.890764 | 5.83 | 0.61 | 1.51% |
| 17 | 0.003448 | 0.047951 | 0.006882 | 0.620062 | 5.75 | 0.63 | 0.86% |
| 18 | 0.035395 | 0.040802 | 0.005528 | 0.819616 | 5.75 | 0.62 | 1.01% |
| 19 | 0.012839 | 0.107622 | 0.009665 | 0.760583 | 5.75 | 0.53 | 0.87% |
| 20 | 0.116419 | 1.038797 | 0.008982 | 1.134517 | 5.75 | 0.60 | 1.83% |
| 21 | 0.004245 | 0.046041 | 0.006043 | 0.639525 | 5.75 | 0.57 | 0.61% |
| 22 | 0.029894 | 0.187537 | 0.006803 | 0.765794 | 5.73 | 0.59 | 1.02% |
| 23 | 0.007997 | 0.045704 | 0.005819 | 0.701728 | 5.72 | 0.62 | 0.71% |
| 24 | 0.003389 | 0.047329 | 0.008168 | 0.703349 | 5.73 | 0.55 | 0.80% |
| 25 | 0.004607 | 0.050169 | 0.006093 | 0.659143 | 5.73 | 0.48 | 0.69% |
| 26 | 0.008074 | 0.057763 | 0.006451 | 0.714662 | 5.77 | 0.49 | 0.62% |
| 27 | 0.009783 | 0.044390 | 0.007746 | 0.743275 | 5.73 | 0.54 | 0.64% |
| 28 | 0.003242 | 0.035908 | 0.005891 | 0.640437 | 5.73 | 0.56 | 0.70% |
| 29 | 0.006425 | 0.039587 | 0.007439 | 0.690141 | 5.77 | 0.62 | 0.70% |
| 30 | 0.078080 | 0.768853 | 0.008977 | 1.084500 | 5.73 | 0.61 | 1.29% |
| | | | | | | | |
| Best | 0.003242 | 0.035908 | 0.005507 | 0.620062 | 5.72 | 0.48 | 0.60% |
| Worst | 0.159072 | 1.038797 | 0.012773 | 1.134517 | 5.83 | 0.63 | 2.77% |
| Average | 0.030687 | 0.178657 | 0.007669 | 0.797626 | 5.75 | 0.57 | 1.00% |
| Median | 0.013646 | 0.049715 | 0.006843 | 0.763189 | 5.75 | 0.58 | 0.86% |
| Std.Dev. | 0.036568 | 0.247334 | 0.001934 | 0.144896 | 0.03 | 0.05 | 0.46% |

Table A.5 – RSearch Statistics for Test Problem 1

| Run | GD | SP | iGD | DI | PT | DR | DD |
|---|---|---|---|---|---|---|---|
| 1 | 0.278377 | 0.287194 | 0.011930 | 1.115569 | 11.66 | 0.93 | 3.59% |
| 2 | 0.025524 | 0.138548 | 0.011315 | 0.774462 | 11.33 | 0.89 | 1.75% |
| 3 | 0.205154 | 1.101590 | 0.011649 | 1.056082 | 11.36 | 0.94 | 2.96% |
| 4 | 0.196355 | 0.668852 | 0.013214 | 0.932668 | 11.23 | 0.89 | 3.53% |
| 5 | 0.181628 | 0.998276 | 0.009714 | 0.946636 | 11.38 | 0.83 | 2.86% |
| 6 | 0.368192 | 0.950191 | 0.012869 | 1.027655 | 11.80 | 0.77 | 6.90% |
| 7 | 0.131483 | 0.870226 | 0.013482 | 1.057247 | 12.17 | 0.83 | 2.11% |
| 8 | 0.322836 | 1.683992 | 0.010941 | 1.132118 | 11.45 | 0.73 | 4.15% |
| 9 | 0.042281 | 0.122810 | 0.011965 | 0.864689 | 11.39 | 0.84 | 1.78% |
| 10 | 0.375358 | 1.144755 | 0.013247 | 1.067966 | 11.22 | 0.86 | 5.41% |
| 11 | 0.334922 | 1.719750 | 0.012178 | 1.091830 | 11.05 | 0.90 | 3.83% |
| 12 | 0.274517 | 1.338292 | 0.012241 | 1.084068 | 11.39 | 0.91 | 3.43% |
| 13 | 0.362665 | 0.977850 | 0.008950 | 1.080673 | 11.50 | 0.84 | 5.12% |
| 14 | 0.145247 | 0.792872 | 0.010086 | 0.914634 | 11.62 | 0.89 | 2.39% |
| 15 | 0.149294 | 0.580615 | 0.010300 | 0.889197 | 11.53 | 0.81 | 2.75% |
| 16 | 0.206578 | 0.985876 | 0.011590 | 1.020879 | 11.52 | 0.86 | 3.31% |
| 17 | 0.005023 | 0.137246 | 0.011371 | 0.698981 | 11.05 | 0.84 | 1.13% |
| 18 | 0.025910 | 0.166851 | 0.010258 | 0.708288 | 11.31 | 0.82 | 1.46% |
| 19 | 0.248197 | 0.603115 | 0.011309 | 1.005179 | 11.50 | 0.84 | 3.91% |
| 20 | 0.065895 | 0.176841 | 0.010649 | 0.801941 | 11.59 | 0.75 | 1.90% |
| 21 | 0.117509 | 0.288631 | 0.014253 | 0.915020 | 11.52 | 0.86 | 2.46% |
| 22 | 0.111876 | 0.418413 | 0.009680 | 0.883223 | 11.48 | 0.88 | 2.15% |
| 23 | 0.027480 | 0.180792 | 0.009769 | 0.757412 | 11.72 | 0.88 | 1.66% |
| 24 | 0.255795 | 1.920278 | 0.010348 | 1.139186 | 11.69 | 0.81 | 3.23% |
| 25 | 0.185649 | 1.176498 | 0.013830 | 1.013320 | 11.02 | 0.77 | 2.81% |
| 26 | 0.152178 | 0.191317 | 0.014914 | 0.934513 | 11.38 | 0.73 | 2.77% |
| 27 | 0.275575 | 0.171258 | 0.011169 | 0.961057 | 11.20 | 0.87 | 4.52% |
| 28 | 0.161210 | 0.599980 | 0.012366 | 0.915304 | 11.27 | 0.83 | 2.66% |
| 29 | 0.045379 | 0.330292 | 0.011309 | 0.834332 | 11.42 | 0.76 | 1.78% |
| 30 | 0.010785 | 0.143129 | 0.012988 | 0.749951 | 11.27 | 0.92 | 1.66% |
| Best | 0.005023 | 0.122810 | 0.008950 | 0.698981 | 11.02 | 0.73 | 1.13% |
| Worst | 0.375358 | 1.920278 | 0.014914 | 1.139186 | 12.17 | 0.94 | 6.90% |
| Average | 0.176296 | 0.695544 | 0.011663 | 0.945803 | 11.43 | 0.84 | 3.00% |
| Median | 0.171419 | 0.601548 | 0.011480 | 0.940574 | 11.41 | 0.84 | 2.79% |
| Std.Dev. | 0.115240 | 0.527502 | 0.001481 | 0.130018 | 0.24 | 0.06 | 1.30% |

Table A.6 – MOPSO Statistics for Test Problem 2

| Run | GD | SP | iGD | DI | PT | DR | DD |
|---|---|---|---|---|---|---|---|
| 1 | 0.008505 | 0.048904 | 0.004963 | 0.753786 | 82.42 | 0.01 | 0.00% |
| 2 | 0.008102 | 0.028250 | 0.007215 | 0.755591 | 88.58 | 0.01 | 0.00% |
| 3 | 0.008227 | 0.032286 | 0.004111 | 0.766736 | 83.86 | 0.01 | 0.00% |
| 4 | 0.007674 | 0.039383 | 0.003723 | 0.689305 | 57.50 | 0.01 | 0.00% |
| 5 | 0.007347 | 0.031549 | 0.004762 | 0.692754 | 61.95 | 0.01 | 0.00% |
| 6 | 0.006670 | 0.047966 | 0.006114 | 0.789779 | 89.66 | 0.02 | 0.00% |
| 7 | 0.171931 | 0.050665 | 0.033589 | 0.838352 | 107.47 | 1.00 | 20.49% |
| 8 | 0.007565 | 0.033218 | 0.004160 | 0.719158 | 83.44 | 0.00 | 0.00% |
| 9 | 0.008179 | 0.030339 | 0.004940 | 0.806589 | 83.50 | 0.01 | 0.00% |
| 10 | 0.146926 | 0.068648 | 0.034006 | 0.843198 | 129.48 | 1.00 | 19.31% |
| 11 | 0.008202 | 0.037945 | 0.004844 | 0.802150 | 88.64 | 0.01 | 0.00% |
| 12 | 0.007575 | 0.033689 | 0.005241 | 0.791521 | 69.77 | 0.02 | 0.05% |
| 13 | 0.186880 | 0.055507 | 0.033764 | 0.791948 | 109.80 | 1.00 | 21.26% |
| 14 | 0.008109 | 0.037717 | 0.003796 | 0.793321 | 88.30 | 0.01 | 0.00% |
| 15 | 0.270149 | 0.156542 | 0.042583 | 1.043060 | 34.16 | 0.99 | 26.72% |
| 16 | 0.007433 | 0.040210 | 0.007002 | 0.786160 | 78.95 | 0.02 | 0.00% |
| 17 | 0.008002 | 0.034512 | 0.004297 | 0.687735 | 88.89 | 0.01 | 0.00% |
| 18 | 0.166359 | 0.058733 | 0.034872 | 0.797714 | 123.81 | 1.00 | 21.43% |
| 19 | 0.167231 | 0.049030 | 0.061793 | 0.796103 | 88.70 | 1.00 | 20.47% |
| 20 | 0.009248 | 0.033774 | 0.008315 | 0.755354 | 87.14 | 0.01 | 0.00% |
| 21 | 0.006737 | 0.057376 | 0.004022 | 0.787616 | 89.48 | 0.01 | 0.00% |
| 22 | 0.164113 | 0.045699 | 0.033585 | 0.810999 | 112.81 | 1.00 | 20.39% |
| 23 | 0.154361 | 0.053068 | 0.033749 | 0.782903 | 126.91 | 1.00 | 19.38% |
| 24 | 0.171778 | 0.060758 | 0.038282 | 0.823206 | 108.33 | 1.00 | 20.76% |
| 25 | 0.008539 | 0.033467 | 0.025303 | 0.798803 | 85.02 | 0.01 | 0.00% |
| 26 | 0.008589 | 0.046421 | 0.004729 | 0.720528 | 87.38 | 0.02 | 0.00% |
| 27 | 0.007935 | 0.037333 | 0.004106 | 0.749669 | 63.12 | 0.00 | 0.00% |
| 28 | 0.160509 | 0.071070 | 0.042124 | 0.852959 | 119.94 | 1.00 | 20.18% |
| 29 | 0.167484 | 0.054078 | 0.034182 | 0.829439 | 118.88 | 1.00 | 20.80% |
| 30 | 0.008061 | 0.045303 | 0.004522 | 0.723988 | 89.80 | 0.02 | 0.00% |
| Best | 0.006670 | 0.028250 | 0.003723 | 0.687735 | 34.16 | 0.00 | 0.00% |
| Worst | 0.270149 | 0.156542 | 0.061793 | 1.043060 | 129.48 | 1.00 | 26.72% |
| Average | 0.069281 | 0.048448 | 0.017957 | 0.786014 | 90.92 | 0.37 | 7.71% |
| Median | 0.008366 | 0.045501 | 0.006558 | 0.790650 | 88.61 | 0.02 | 0.00% |
| Std.Dev. | 0.084279 | 0.023442 | 0.017021 | 0.066072 | 21.79 | 0.48 | 10.37% |

Table A.7 – MOPSO-e Statistics for Test Problem 2

| Run | GD | SP | iGD | DI | PT | DR | DD |
|---|---|---|---|---|---|---|---|
| 1 | 0.174037 | 0.064036 | 0.109034 | 0.911658 | 102.66 | 0.98 | 23.08% |
| 2 | 0.007818 | 0.040541 | 0.003489 | 0.713981 | 85.28 | 0.01 | 0.00% |
| 3 | 0.007894 | 0.033061 | 0.007337 | 0.720583 | 98.98 | 0.01 | 0.00% |
| 4 | 0.172392 | 0.058853 | 0.061584 | 0.785303 | 138.45 | 1.00 | 20.67% |
| 5 | 0.008706 | 0.041070 | 0.004126 | 0.727595 | 92.55 | 0.01 | 0.00% |
| 6 | 0.007115 | 0.034691 | 0.003503 | 0.794788 | 99.05 | 0.04 | 0.00% |
| 7 | 0.006476 | 0.033936 | 0.005189 | 0.759083 | 100.70 | 0.01 | 0.00% |
| 8 | 0.008139 | 0.043710 | 0.011929 | 0.762105 | 100.34 | 0.01 | 0.00% |
| 9 | 0.179614 | 0.059330 | 0.034258 | 0.834438 | 137.25 | 1.00 | 20.76% |
| 10 | 0.008664 | 0.033302 | 0.004244 | 0.704680 | 101.66 | 0.02 | 0.00% |
| 11 | 0.007977 | 0.048381 | 0.003998 | 0.773068 | 90.44 | 0.02 | 0.00% |
| 12 | 0.007809 | 0.048758 | 0.004854 | 0.777760 | 101.45 | 0.02 | 0.00% |
| 13 | 0.008596 | 0.038970 | 0.004132 | 0.769297 | 97.34 | 0.00 | 0.00% |
| 14 | 0.007833 | 0.038995 | 0.003717 | 0.703473 | 99.53 | 0.01 | 0.00% |
| 15 | 0.009967 | 0.030223 | 0.042342 | 0.709665 | 41.44 | 0.08 | 0.16% |
| 16 | 0.007724 | 0.037946 | 0.003557 | 0.778794 | 92.64 | 0.02 | 0.00% |
| 17 | 0.006430 | 0.033197 | 0.003594 | 0.670272 | 74.55 | 0.01 | 0.00% |
| 18 | 0.191442 | 0.042770 | 0.043728 | 0.827520 | 107.83 | 0.98 | 21.73% |
| 19 | 0.008311 | 0.028308 | 0.010741 | 0.713989 | 98.19 | 0.01 | 0.00% |
| 20 | 0.008010 | 0.033440 | 0.003631 | 0.697921 | 86.41 | 0.02 | 0.00% |
| 21 | 0.007170 | 0.042940 | 0.004572 | 0.831286 | 99.31 | 0.03 | 0.00% |
| 22 | 0.007977 | 0.045430 | 0.009918 | 0.760834 | 90.66 | 0.01 | 0.00% |
| 23 | 0.152928 | 0.095670 | 0.066510 | 0.870544 | 132.64 | 1.00 | 20.03% |
| 24 | 0.166308 | 0.065334 | 0.034630 | 0.826427 | 134.56 | 1.00 | 20.29% |
| 25 | 0.007867 | 0.042031 | 0.012382 | 0.900211 | 103.19 | 0.01 | 0.00% |
| 26 | 0.008080 | 0.055533 | 0.024725 | 0.792857 | 63.33 | 0.00 | 0.00% |
| 27 | 0.007624 | 0.036505 | 0.005046 | 0.805990 | 97.59 | 0.04 | 0.00% |
| 28 | 0.202686 | 0.041719 | 0.129372 | 0.888394 | 115.44 | 0.98 | 24.10% |
| 29 | 0.008329 | 0.048219 | 0.003385 | 0.728101 | 92.12 | 0.01 | 0.00% |
| 30 | 0.007792 | 0.036212 | 0.007314 | 0.711664 | 99.91 | 0.02 | 0.00% |
| Best | 0.006430 | 0.028308 | 0.003385 | 0.670272 | 41.44 | 0.00 | 0.00% |
| Worst | 0.202686 | 0.095670 | 0.129372 | 0.911658 | 138.45 | 1.00 | 24.10% |
| Average | 0.047391 | 0.044437 | 0.022228 | 0.775076 | 99.18 | 0.25 | 5.03% |
| Median | 0.008045 | 0.041394 | 0.006252 | 0.771183 | 99.18 | 0.02 | 0.00% |
| Std.Dev. | 0.073138 | 0.013763 | 0.031937 | 0.064595 | 20.02 | 0.42 | 9.28% |

Table A.8 – MOPSO-p Statistics for Test Problem 2

| Run | GD | SP | iGD | DI | PT | DR | DD |
|---|---|---|---|---|---|---|---|
| 1 | 0.007858 | 0.034587 | 0.003693 | 0.729661 | 91.56 | 0.02 | 0.00% |
| 2 | 0.008272 | 0.039101 | 0.004717 | 0.793260 | 85.16 | 0.03 | 0.00% |
| 3 | 0.154367 | 0.064618 | 0.038202 | 0.850103 | 97.94 | 1.00 | 19.46% |
| 4 | 0.008871 | 0.036377 | 0.012073 | 0.857694 | 89.17 | 0.01 | 0.00% |
| 5 | 0.008130 | 0.034550 | 0.003342 | 0.693122 | 88.27 | 0.01 | 0.00% |
| 6 | 0.139260 | 0.063062 | 0.053530 | 0.899281 | 128.16 | 1.00 | 19.44% |
| 7 | 0.007641 | 0.042392 | 0.004073 | 0.736321 | 88.98 | 0.02 | 0.00% |
| 8 | 0.179426 | 0.063933 | 0.037151 | 0.800216 | 125.47 | 1.00 | 20.97% |
| 9 | 0.139903 | 0.082946 | 0.037531 | 0.834987 | 88.69 | 1.00 | 19.06% |
| 10 | 0.007731 | 0.038601 | 0.007887 | 0.739471 | 87.66 | 0.01 | 0.00% |
| 11 | 0.008115 | 0.050881 | 0.004583 | 0.713145 | 88.22 | 0.02 | 0.00% |
| 12 | 0.006832 | 0.056882 | 0.003906 | 0.785044 | 86.45 | 0.02 | 0.00% |
| 13 | 0.146549 | 0.063088 | 0.038583 | 0.860842 | 125.64 | 1.00 | 19.23% |
| 14 | 0.007635 | 0.035107 | 0.003821 | 0.740534 | 62.19 | 0.03 | 0.01% |
| 15 | 0.007124 | 0.040273 | 0.003728 | 0.788018 | 67.77 | 0.02 | 0.00% |
| 16 | 0.008287 | 0.040638 | 0.005677 | 0.861766 | 87.73 | 0.02 | 0.00% |
| 17 | 0.007466 | 0.034986 | 0.003911 | 0.777695 | 87.42 | 0.02 | 0.00% |
| 18 | 0.007761 | 0.039153 | 0.004656 | 0.831372 | 86.20 | 0.02 | 0.00% |
| 19 | 0.168381 | 0.046766 | 0.034294 | 0.884615 | 83.92 | 1.00 | 21.05% |
| 20 | 0.145215 | 0.065533 | 0.033934 | 0.810535 | 92.84 | 1.00 | 20.02% |
| 21 | 0.165073 | 0.054138 | 0.072760 | 0.829764 | 127.23 | 1.00 | 19.80% |
| 22 | 0.008197 | 0.039160 | 0.004234 | 0.733938 | 76.03 | 0.01 | 0.00% |
| 23 | 0.008790 | 0.036028 | 0.009830 | 0.697783 | 63.59 | 0.02 | 0.01% |
| 24 | 0.007951 | 0.036303 | 0.004787 | 0.757319 | 88.08 | 0.02 | 0.00% |
| 25 | 0.155244 | 0.062588 | 0.035866 | 0.803205 | 119.12 | 1.00 | 20.36% |
| 26 | 0.008065 | 0.035291 | 0.003486 | 0.719925 | 77.47 | 0.01 | 0.00% |
| 27 | 0.008649 | 0.040946 | 0.018341 | 0.707581 | 88.50 | 0.01 | 0.00% |
| 28 | 0.159528 | 0.067323 | 0.034948 | 0.815549 | 107.94 | 1.00 | 20.60% |
| 29 | 0.156012 | 0.072946 | 0.037761 | 0.833642 | 120.83 | 1.00 | 20.29% |
| 30 | 0.007174 | 0.046478 | 0.004635 | 0.737001 | 84.28 | 0.02 | 0.00% |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Best | 0.006832 | 0.034550 | 0.003342 | 0.693122 | 62.19 | 0.01 | 0.00% |
| Worst | 0.179426 | 0.082946 | 0.072760 | 0.899281 | 128.16 | 1.00 | 21.05% |
| Average | 0.061984 | 0.048822 | 0.018865 | 0.787446 | 93.08 | 0.38 | 7.34% |
| Median | 0.008280 | 0.041669 | 0.006782 | 0.790639 | 88.25 | 0.02 | 0.00% |
| Std.Dev. | 0.072631 | 0.013924 | 0.018932 | 0.059268 | 18.34 | 0.48 | 9.82% |

Table A.9 – NSGA-II Statistics for Test Problem 2

| Run | GD | SP | iGD | DI | PT | DR | DD |
|---|---|---|---|---|---|---|---|
| 1 | 0.007272 | 0.036953 | 0.001993 | 0.602407 | 13.86 | 0.05 | 0.03% |
| 2 | 0.161663 | 0.050215 | 0.033506 | 0.703803 | 14.36 | 1.00 | 20.25% |
| 3 | 0.007738 | 0.030602 | 0.001828 | 0.549361 | 13.97 | 0.02 | 0.01% |
| 4 | 0.071726 | 0.045471 | 0.002368 | 1.024889 | 13.73 | 0.20 | 4.06% |
| 5 | 0.006972 | 0.027859 | 0.002499 | 0.603976 | 13.77 | 0.03 | 0.03% |
| 6 | 0.007302 | 0.034688 | 0.002354 | 0.550586 | 13.80 | 0.04 | 0.06% |
| 7 | 0.007200 | 0.034988 | 0.002185 | 0.599569 | 13.78 | 0.02 | 0.00% |
| 8 | 0.007218 | 0.035960 | 0.002074 | 0.583587 | 13.80 | 0.04 | 0.04% |
| 9 | 0.008004 | 0.032970 | 0.002225 | 0.561710 | 13.78 | 0.01 | 0.00% |
| 10 | 0.007298 | 0.032000 | 0.002306 | 0.581572 | 13.78 | 0.01 | 0.00% |
| 11 | 0.159050 | 0.061617 | 0.033493 | 0.768659 | 13.89 | 1.00 | 19.77% |
| 12 | 0.097806 | 0.752934 | 0.002726 | 1.148299 | 13.80 | 0.06 | 15.45% |
| 13 | 0.007506 | 0.031273 | 0.001952 | 0.539223 | 13.81 | 0.02 | 0.00% |
| 14 | 0.007741 | 0.035590 | 0.002512 | 0.610800 | 13.80 | 0.04 | 0.01% |
| 15 | 0.151299 | 0.060447 | 0.033463 | 0.766471 | 13.91 | 1.00 | 19.31% |
| 16 | 0.007653 | 0.035908 | 0.002177 | 0.570681 | 13.78 | 0.01 | 0.00% |
| 17 | 0.008059 | 0.041417 | 0.001968 | 0.636743 | 13.81 | 0.04 | 0.04% |
| 18 | 0.006736 | 0.039928 | 0.002219 | 0.578906 | 13.80 | 0.03 | 0.03% |
| 19 | 0.007090 | 0.031490 | 0.002410 | 0.579994 | 13.77 | 0.05 | 0.02% |
| 20 | 0.097037 | 0.964998 | 0.002707 | 1.169907 | 13.81 | 0.06 | 12.74% |
| 21 | 0.007394 | 0.031051 | 0.002274 | 0.579280 | 13.78 | 0.04 | 0.01% |
| 22 | 0.157191 | 0.053967 | 0.033464 | 0.754219 | 13.91 | 1.00 | 19.54% |
| 23 | 0.007235 | 0.035005 | 0.002079 | 0.611786 | 13.77 | 0.02 | 0.15% |
| 24 | 0.008024 | 0.033560 | 0.002186 | 0.584432 | 13.81 | 0.01 | 0.00% |
| 25 | 0.007499 | 0.036767 | 0.002127 | 0.605853 | 13.78 | 0.05 | 0.01% |
| 26 | 0.095664 | 0.075649 | 0.003281 | 1.088957 | 13.75 | 0.04 | 26.55% |
| 27 | 0.148585 | 0.063823 | 0.033450 | 0.743571 | 13.89 | 1.00 | 19.37% |
| 28 | 0.006908 | 0.037081 | 0.001699 | 0.591181 | 13.78 | 0.09 | 0.05% |
| 29 | 0.006651 | 0.037173 | 0.001845 | 0.610589 | 13.78 | 0.03 | 0.28% |
| 30 | 0.049922 | 0.492096 | 0.002805 | 1.051519 | 13.78 | 0.02 | 19.47% |
| Best | 0.006651 | 0.027859 | 0.001699 | 0.539223 | 13.73 | 0.01 | 0.00% |
| Worst | 0.161663 | 0.964998 | 0.033506 | 1.169907 | 14.36 | 1.00 | 26.55% |
| Average | 0.044581 | 0.110449 | 0.007472 | 0.698418 | 13.83 | 0.20 | 5.91% |
| Median | 0.007695 | 0.036860 | 0.002290 | 0.604915 | 13.80 | 0.04 | 0.04% |
| Std.Dev. | 0.058234 | 0.221534 | 0.011832 | 0.193396 | 0.11 | 0.37 | 9.07% |

Table A.10 – RSearch Statistics for Test Problem 2

| Run | GD | SP | iGD | DI | PT | DR | DD |
|---|---|---|---|---|---|---|---|
| 1 | 0.323687 | 0.185961 | 0.010469 | 1.141194 | 11.50 | 0.70 | 18.69% |
| 2 | 0.283155 | 0.252599 | 0.017429 | 1.201251 | 12.14 | 0.83 | 20.61% |
| 3 | 0.354423 | 0.114325 | 0.009315 | 1.112527 | 11.27 | 0.65 | 23.10% |
| 4 | 0.317702 | 0.219610 | 0.010424 | 1.135544 | 11.09 | 0.68 | 17.55% |
| 5 | 0.291232 | 0.176857 | 0.010078 | 1.083162 | 11.64 | 0.65 | 17.37% |
| 6 | 0.251447 | 0.355268 | 0.008139 | 1.070235 | 11.48 | 0.70 | 7.90% |
| 7 | 0.361728 | 0.464826 | 0.009777 | 1.145521 | 9.97 | 0.68 | 11.77% |
| 8 | 0.244219 | 0.384002 | 0.010862 | 1.209031 | 11.09 | 0.61 | 8.76% |
| 9 | 0.226331 | 0.293125 | 0.009253 | 1.107292 | 10.36 | 0.62 | 8.22% |
| 10 | 0.294910 | 0.505319 | 0.007946 | 1.149054 | 11.02 | 0.69 | 12.78% |
| 11 | 0.283388 | 0.177290 | 0.012153 | 1.090528 | 11.88 | 0.68 | 13.13% |
| 12 | 0.271825 | 0.228778 | 0.008818 | 1.110695 | 10.64 | 0.64 | 11.01% |
| 13 | 0.208477 | 0.113288 | 0.007089 | 1.054119 | 10.44 | 0.63 | 6.76% |
| 14 | 0.343095 | 0.259833 | 0.014770 | 1.137510 | 11.80 | 0.63 | 20.00% |
| 15 | 0.364924 | 0.502576 | 0.008545 | 1.091814 | 10.50 | 0.70 | 13.38% |
| 16 | 0.275634 | 0.258197 | 0.012496 | 1.188576 | 10.55 | 0.63 | 8.34% |
| 17 | 0.308842 | 0.358232 | 0.011325 | 1.192930 | 11.39 | 0.85 | 7.20% |
| 18 | 0.367636 | 0.611342 | 0.012405 | 1.204520 | 11.02 | 0.58 | 18.93% |
| 19 | 0.278929 | 0.117707 | 0.009976 | 1.160925 | 11.48 | 0.67 | 13.88% |
| 20 | 0.297949 | 0.485120 | 0.012540 | 1.147134 | 11.17 | 0.66 | 11.57% |
| 21 | 0.294473 | 0.157960 | 0.013420 | 0.996884 | 11.64 | 0.77 | 13.95% |
| 22 | 0.349482 | 0.497245 | 0.016448 | 1.174928 | 10.30 | 0.66 | 14.56% |
| 23 | 0.321586 | 0.158741 | 0.011073 | 1.100435 | 11.47 | 0.79 | 19.30% |
| 24 | 0.224874 | 0.331643 | 0.012316 | 1.163866 | 11.08 | 0.68 | 7.71% |
| 25 | 0.336104 | 0.173988 | 0.009995 | 1.108183 | 10.64 | 0.79 | 14.07% |
| 26 | 0.274223 | 0.077121 | 0.008858 | 1.154245 | 11.22 | 0.68 | 13.80% |
| 27 | 0.278278 | 0.187584 | 0.009467 | 1.150764 | 11.16 | 0.50 | 14.45% |
| 28 | 0.297560 | 0.148884 | 0.009517 | 1.112092 | 12.66 | 0.68 | 15.05% |
| 29 | 0.325805 | 0.643191 | 0.008492 | 1.164127 | 10.75 | 0.67 | 13.82% |
| 30 | 0.281372 | 0.535984 | 0.008426 | 1.150775 | 12.17 | 0.68 | 9.08% |
| Best | 0.208477 | 0.077121 | 0.007089 | 0.996884 | 9.97 | 0.50 | 6.76% |
| Worst | 0.367636 | 0.643191 | 0.017429 | 1.209031 | 12.66 | 0.85 | 23.10% |
| Average | 0.297776 | 0.299220 | 0.010727 | 1.133662 | 11.18 | 0.68 | 13.56% |
| Median | 0.294691 | 0.255398 | 0.010037 | 1.143357 | 11.17 | 0.68 | 13.81% |
| Std.Dev. | 0.042362 | 0.163218 | 0.002458 | 0.048007 | 0.62 | 0.07 | 4.44% |

Table A.11 – MOPSO Statistics for Test Problem 3

| Run | GD | SP | iGD | DI | PT | DR | DD |
|---|---|---|---|---|---|---|---|
| 1 | 0.006352 | 0.087460 | 0.009390 | 0.727657 | 22.33 | 0.57 | 0.85% |
| 2 | 0.005638 | 0.130211 | 0.008756 | 0.756899 | 21.97 | 0.58 | 0.64% |
| 3 | 0.005657 | 0.115268 | 0.008045 | 0.658388 | 21.16 | 0.58 | 1.20% |
| 4 | 0.007370 | 0.090423 | 0.010918 | 0.684292 | 24.38 | 0.62 | 0.93% |
| 5 | 0.005570 | 0.127947 | 0.008252 | 0.740068 | 23.67 | 0.59 | 0.74% |
| 6 | 0.006010 | 0.111105 | 0.018447 | 0.688761 | 21.42 | 0.56 | 1.15% |
| 7 | 0.006887 | 0.064576 | 0.012193 | 0.675527 | 22.69 | 0.61 | 1.02% |
| 8 | 0.005670 | 0.114315 | 0.011090 | 0.759070 | 23.39 | 0.67 | 1.04% |
| 9 | 0.005588 | 0.115541 | 0.009753 | 0.710397 | 19.66 | 0.53 | 0.53% |
| 10 | 0.005689 | 0.085078 | 0.008235 | 0.676740 | 20.75 | 0.64 | 1.33% |
| 11 | 0.006548 | 0.104306 | 0.007922 | 0.657323 | 20.50 | 0.58 | 1.29% |
| 12 | 0.004987 | 0.116938 | 0.007084 | 0.693551 | 22.39 | 0.67 | 0.69% |
| 13 | 0.004605 | 0.114308 | 0.007258 | 0.679830 | 21.67 | 0.59 | 0.42% |
| 14 | 0.005546 | 0.123283 | 0.009594 | 0.731295 | 21.62 | 0.56 | 0.96% |
| 15 | 0.005814 | 0.133643 | 0.008121 | 0.639670 | 22.38 | 0.61 | 0.75% |
| 16 | 0.005157 | 0.118396 | 0.008770 | 0.694003 | 23.23 | 0.65 | 0.80% |
| 17 | 0.005664 | 0.072568 | 0.010638 | 0.711539 | 20.80 | 0.56 | 0.73% |
| 18 | 0.005308 | 0.125995 | 0.008451 | 0.719181 | 21.25 | 0.54 | 1.01% |
| 19 | 0.005686 | 0.113161 | 0.009214 | 0.696151 | 21.48 | 0.58 | 0.86% |
| 20 | 0.004914 | 0.120936 | 0.008241 | 0.757497 | 19.89 | 0.52 | 0.97% |
| 21 | 0.006011 | 0.118600 | 0.009311 | 0.734720 | 18.59 | 0.67 | 0.68% |
| 22 | 0.005203 | 0.111231 | 0.008590 | 0.691011 | 22.23 | 0.59 | 0.77% |
| 23 | 0.005179 | 0.108861 | 0.011423 | 0.733364 | 22.84 | 0.62 | 0.82% |
| 24 | 0.006965 | 0.112885 | 0.010460 | 0.695160 | 21.36 | 0.62 | 0.85% |
| 25 | 0.004436 | 0.164352 | 0.011016 | 0.739412 | 20.72 | 0.47 | 0.36% |
| 26 | 0.005697 | 0.123769 | 0.009026 | 0.747390 | 23.06 | 0.46 | 0.92% |
| 27 | 0.005293 | 0.115942 | 0.007437 | 0.665133 | 22.30 | 0.62 | 0.72% |
| 28 | 0.004973 | 0.111891 | 0.007411 | 0.691807 | 20.80 | 0.63 | 0.50% |
| 29 | 0.005304 | 0.121838 | 0.009870 | 0.693458 | 17.98 | 0.54 | 1.00% |
| 30 | 0.007153 | 0.079611 | 0.010276 | 0.711646 | 23.16 | 0.53 | 0.87% |
| | | | | | | | |
| Best | 0.004436 | 0.064576 | 0.007084 | 0.639670 | 17.98 | 0.46 | 0.36% |
| Worst | 0.007370 | 0.164352 | 0.018447 | 0.759070 | 24.38 | 0.67 | 1.33% |
| Average | 0.005696 | 0.111815 | 0.009506 | 0.705365 | 21.66 | 0.59 | 0.85% |
| Median | 0.005648 | 0.114792 | 0.009120 | 0.695655 | 21.65 | 0.58 | 0.85% |
| Std.Dev. | 0.000721 | 0.019747 | 0.002157 | 0.032172 | 1.45 | 0.05 | 0.23% |

Table A.12 – MOPSO-e Statistics for Test Problem 3

| Run | GD | SP | iGD | DI | PT | DR | DD |
|---|---|---|---|---|---|---|---|
| 1 | 0.005284 | 0.128873 | 0.007929 | 0.719520 | 29.42 | 0.55 | 0.75% |
| 2 | 0.005357 | 0.110243 | 0.009060 | 0.805355 | 30.28 | 0.60 | 0.66% |
| 3 | 0.006497 | 0.136546 | 0.008751 | 0.700338 | 30.98 | 0.51 | 0.94% |
| 4 | 0.004503 | 0.122275 | 0.007657 | 0.694745 | 28.05 | 0.58 | 0.66% |
| 5 | 0.004982 | 0.147194 | 0.008397 | 0.714777 | 30.08 | 0.57 | 0.70% |
| 6 | 0.005471 | 0.129823 | 0.008082 | 0.708260 | 29.98 | 0.53 | 0.82% |
| 7 | 0.004705 | 0.151244 | 0.010864 | 0.757818 | 28.53 | 0.53 | 0.54% |
| 8 | 0.004276 | 0.207753 | 0.013142 | 0.802809 | 28.73 | 0.54 | 0.46% |
| 9 | 0.004995 | 0.130867 | 0.007144 | 0.692018 | 30.72 | 0.56 | 0.65% |
| 10 | 0.004845 | 0.143545 | 0.008424 | 0.742552 | 26.91 | 0.55 | 0.77% |
| 11 | 0.004959 | 0.117464 | 0.008006 | 0.830798 | 28.02 | 0.55 | 0.54% |
| 12 | 0.005781 | 0.122167 | 0.012022 | 0.668814 | 29.86 | 0.56 | 0.91% |
| 13 | 0.004939 | 0.147238 | 0.007170 | 0.750431 | 27.39 | 0.53 | 0.67% |
| 14 | 0.004455 | 0.139824 | 0.007891 | 0.728566 | 29.97 | 0.52 | 0.65% |
| 15 | 0.004958 | 0.140201 | 0.009072 | 0.783132 | 30.31 | 0.52 | 0.77% |
| 16 | 0.006864 | 0.130185 | 0.010014 | 0.716546 | 27.27 | 0.61 | 1.03% |
| 17 | 0.005608 | 0.132917 | 0.008403 | 0.691367 | 26.94 | 0.57 | 0.69% |
| 18 | 0.004569 | 0.123801 | 0.008703 | 0.724937 | 26.75 | 0.47 | 0.61% |
| 19 | 0.004513 | 0.070288 | 0.010971 | 0.625549 | 29.59 | 0.66 | 0.38% |
| 20 | 0.004696 | 0.140535 | 0.008696 | 0.766018 | 29.17 | 0.52 | 0.34% |
| 21 | 0.005078 | 0.122576 | 0.008016 | 0.720867 | 28.67 | 0.64 | 0.65% |
| 22 | 0.005063 | 0.147660 | 0.008211 | 0.665884 | 28.77 | 0.64 | 0.74% |
| 23 | 0.005489 | 0.056757 | 0.012376 | 0.744782 | 30.70 | 0.64 | 0.54% |
| 24 | 0.005956 | 0.125892 | 0.008374 | 0.710576 | 29.52 | 0.62 | 0.69% |
| 25 | 0.006942 | 0.080377 | 0.013389 | 0.723887 | 31.03 | 0.47 | 1.27% |
| 26 | 0.006335 | 0.132695 | 0.009275 | 0.657069 | 29.56 | 0.52 | 0.81% |
| 27 | 0.004604 | 0.111815 | 0.008588 | 0.737277 | 28.69 | 0.62 | 0.51% |
| 28 | 0.005686 | 0.133131 | 0.009518 | 0.801791 | 28.28 | 0.53 | 0.79% |
| 29 | 0.004727 | 0.134105 | 0.008489 | 0.748097 | 27.77 | 0.55 | 0.69% |
| 30 | 0.007423 | 0.087472 | 0.011666 | 0.742153 | 29.88 | 0.60 | 0.95% |
| | | | | | | | |
| Best | 0.004276 | 0.056757 | 0.007144 | 0.625549 | 26.75 | 0.47 | 0.34% |
| Worst | 0.007423 | 0.207753 | 0.013389 | 0.830798 | 31.03 | 0.66 | 1.27% |
| Average | 0.005319 | 0.126849 | 0.009277 | 0.729224 | 29.06 | 0.56 | 0.71% |
| Median | 0.005029 | 0.130526 | 0.008642 | 0.724412 | 29.30 | 0.55 | 0.69% |
| Std.Dev. | 0.000811 | 0.027541 | 0.001732 | 0.046666 | 1.27 | 0.05 | 0.19% |

Table A.13 – MOPSO-p Statistics for Test Problem 3

| Run | GD | SP | iGD | DI | PT | DR | DD |
|---|---|---|---|---|---|---|---|
| 1 | 0.004473 | 0.105865 | 0.007867 | 0.721744 | 24.42 | 0.63 | 0.67% |
| 2 | 0.005992 | 0.101969 | 0.008357 | 0.739483 | 23.14 | 0.45 | 0.65% |
| 3 | 0.004989 | 0.075203 | 0.011181 | 0.704195 | 23.77 | 0.44 | 1.00% |
| 4 | 0.004863 | 0.161343 | 0.009112 | 0.787958 | 23.00 | 0.52 | 0.56% |
| 5 | 0.005153 | 0.086966 | 0.007718 | 0.703133 | 23.12 | 0.51 | 1.04% |
| 6 | 0.004475 | 0.123344 | 0.007971 | 0.717326 | 24.00 | 0.51 | 0.51% |
| 7 | 0.005118 | 0.068766 | 0.007943 | 0.738956 | 21.00 | 0.62 | 0.68% |
| 8 | 0.006866 | 0.094125 | 0.007699 | 0.631958 | 24.19 | 0.55 | 0.94% |
| 9 | 0.005731 | 0.101204 | 0.010940 | 0.722127 | 24.39 | 0.47 | 0.72% |
| 10 | 0.007708 | 0.087542 | 0.009265 | 0.725611 | 23.64 | 0.47 | 1.44% |
| 11 | 0.004572 | 0.092028 | 0.007800 | 0.712802 | 25.20 | 0.58 | 0.50% |
| 12 | 0.005505 | 0.115798 | 0.010034 | 0.733971 | 21.88 | 0.56 | 0.68% |
| 13 | 0.007830 | 0.083126 | 0.010259 | 0.719083 | 23.08 | 0.54 | 1.27% |
| 14 | 0.006151 | 0.106953 | 0.009325 | 0.740717 | 23.75 | 0.56 | 0.93% |
| 15 | 0.005725 | 0.099499 | 0.010749 | 0.742987 | 24.19 | 0.55 | 1.00% |
| 16 | 0.005019 | 0.110052 | 0.008075 | 0.675130 | 24.08 | 0.58 | 0.99% |
| 17 | 0.006322 | 0.080258 | 0.010191 | 0.651561 | 24.55 | 0.55 | 1.14% |
| 18 | 0.004400 | 0.077113 | 0.008015 | 0.746759 | 22.75 | 0.60 | 0.73% |
| 19 | 0.006254 | 0.104798 | 0.008101 | 0.676103 | 22.25 | 0.56 | 0.78% |
| 20 | 0.004742 | 0.127094 | 0.009105 | 0.755104 | 22.61 | 0.50 | 0.49% |
| 21 | 0.004576 | 0.113872 | 0.007352 | 0.726607 | 24.20 | 0.47 | 0.53% |
| 22 | 0.005286 | 0.110378 | 0.007224 | 0.664455 | 25.28 | 0.52 | 1.22% |
| 23 | 0.005328 | 0.168407 | 0.012792 | 0.772434 | 21.47 | 0.68 | 0.67% |
| 24 | 0.005152 | 0.140102 | 0.009398 | 0.746283 | 20.17 | 0.56 | 0.62% |
| 25 | 0.004830 | 0.116351 | 0.008237 | 0.675685 | 20.31 | 0.58 | 0.72% |
| 26 | 0.006678 | 0.065539 | 0.007562 | 0.674341 | 22.62 | 0.51 | 0.66% |
| 27 | 0.004869 | 0.076819 | 0.011213 | 0.763195 | 23.75 | 0.53 | 0.63% |
| 28 | 0.004704 | 0.120071 | 0.008279 | 0.743434 | 21.56 | 0.58 | 0.55% |
| 29 | 0.004939 | 0.106694 | 0.007299 | 0.721747 | 21.12 | 0.57 | 0.83% |
| 30 | 0.006901 | 0.073611 | 0.010191 | 0.695487 | 22.23 | 0.53 | 1.13% |
| Best | 0.004400 | 0.065539 | 0.007224 | 0.631958 | 20.17 | 0.44 | 0.49% |
| Worst | 0.007830 | 0.168407 | 0.012792 | 0.787958 | 25.28 | 0.68 | 1.44% |
| Average | 0.005505 | 0.103163 | 0.008975 | 0.717679 | 23.06 | 0.54 | 0.81% |
| Median | 0.005152 | 0.103384 | 0.008318 | 0.721937 | 23.13 | 0.55 | 0.72% |
| Std.Dev. | 0.000952 | 0.025010 | 0.001437 | 0.036645 | 1.38 | 0.05 | 0.26% |

Table A.14 – NSGA-II Statistics for Test Problem 3

| Run | GD | SP | iGD | DI | PT | DR | DD |
|---|---|---|---|---|---|---|---|
| 1 | 0.003819 | 0.101064 | 0.004384 | 0.447639 | 17.34 | 0.27 | 0.26% |
| 2 | 0.004074 | 0.095229 | 0.004612 | 0.436469 | 17.23 | 0.23 | 0.65% |
| 3 | 0.003965 | 0.100924 | 0.004922 | 0.502980 | 17.25 | 0.41 | 0.65% |
| 4 | 0.004107 | 0.094204 | 0.004617 | 0.426113 | 17.25 | 0.41 | 0.44% |
| 5 | 0.004161 | 0.094871 | 0.005001 | 0.486655 | 17.27 | 0.34 | 0.72% |
| 6 | 0.004010 | 0.101838 | 0.004646 | 0.466518 | 17.25 | 0.34 | 0.75% |
| 7 | 0.003905 | 0.099409 | 0.004727 | 0.491797 | 17.25 | 0.39 | 0.29% |
| 8 | 0.004112 | 0.108618 | 0.004737 | 0.492351 | 17.23 | 0.34 | 0.43% |
| 9 | 0.003793 | 0.102857 | 0.004893 | 0.497215 | 17.28 | 0.22 | 0.98% |
| 10 | 0.004111 | 0.095226 | 0.004485 | 0.452729 | 17.30 | 0.35 | 0.70% |
| 11 | 0.003681 | 0.099359 | 0.004325 | 0.425481 | 17.27 | 0.33 | 0.26% |
| 12 | 0.004151 | 0.109915 | 0.004873 | 0.503561 | 17.27 | 0.41 | 0.26% |
| 13 | 0.003870 | 0.097185 | 0.005006 | 0.469012 | 17.27 | 0.38 | 0.45% |
| 14 | 0.003926 | 0.102844 | 0.005460 | 0.554743 | 17.28 | 0.26 | 0.22% |
| 15 | 0.004137 | 0.097325 | 0.004734 | 0.448517 | 17.25 | 0.40 | 0.39% |
| 16 | 0.004157 | 0.065049 | 0.004815 | 0.483967 | 17.31 | 0.43 | 0.70% |
| 17 | 0.004215 | 0.101868 | 0.004641 | 0.451538 | 17.25 | 0.37 | 0.51% |
| 18 | 0.003912 | 0.099818 | 0.004772 | 0.461280 | 17.25 | 0.27 | 0.29% |
| 19 | 0.003937 | 0.061826 | 0.005510 | 0.552597 | 17.23 | 0.27 | 0.50% |
| 20 | 0.003969 | 0.056072 | 0.004307 | 0.441752 | 17.28 | 0.32 | 0.24% |
| 21 | 0.004021 | 0.101982 | 0.004761 | 0.484279 | 17.27 | 0.27 | 0.82% |
| 22 | 0.003564 | 0.102235 | 0.004211 | 0.465023 | 17.27 | 0.30 | 0.24% |
| 23 | 0.004158 | 0.100498 | 0.005198 | 0.517794 | 17.27 | 0.36 | 0.68% |
| 24 | 0.003865 | 0.105078 | 0.004605 | 0.515096 | 17.28 | 0.35 | 0.42% |
| 25 | 0.003751 | 0.093342 | 0.004541 | 0.439147 | 17.44 | 0.39 | 0.44% |
| 26 | 0.003929 | 0.101150 | 0.004321 | 0.439300 | 17.67 | 0.29 | 0.49% |
| 27 | 0.003992 | 0.098268 | 0.004903 | 0.457092 | 17.28 | 0.38 | 0.43% |
| 28 | 0.003710 | 0.096392 | 0.004475 | 0.449103 | 17.25 | 0.31 | 0.16% |
| 29 | 0.003849 | 0.094444 | 0.004423 | 0.446391 | 17.28 | 0.29 | 0.51% |
| 30 | 0.004160 | 0.097361 | 0.004894 | 0.447535 | 17.28 | 0.41 | 0.44% |
| | | | | | | | |
| Best | 0.003564 | 0.056072 | 0.004211 | 0.425481 | 17.23 | 0.22 | 0.16% |
| Worst | 0.004215 | 0.109915 | 0.005510 | 0.554743 | 17.67 | 0.43 | 0.98% |
| Average | 0.003967 | 0.095875 | 0.004727 | 0.471789 | 17.29 | 0.34 | 0.48% |
| Median | 0.003967 | 0.099384 | 0.004731 | 0.463152 | 17.27 | 0.34 | 0.44% |
| Std.Dev. | 0.000167 | 0.012528 | 0.000315 | 0.034208 | 0.08 | 0.06 | 0.20% |

Table A.15 – RSearch Statistics for Test Problem 3

| Run | GD | SP | iGD | DI | PT | DR | DD |
|---|---|---|---|---|---|---|---|
| 1 | 0.092114 | 0.409924 | 0.051442 | 0.749208 | 10.95 | 1.00 | 6.61% |
| 2 | 0.077429 | 0.560316 | 0.046308 | 0.613357 | 10.84 | 0.96 | 6.06% |
| 3 | 0.096312 | 0.271394 | 0.051900 | 0.561457 | 10.91 | 1.00 | 6.04% |
| 4 | 0.085096 | 0.693436 | 0.054011 | 0.646170 | 10.84 | 1.00 | 6.40% |
| 5 | 0.097156 | 0.464184 | 0.045506 | 0.567697 | 10.80 | 1.00 | 5.50% |
| 6 | 0.097894 | 0.483251 | 0.056263 | 0.610147 | 10.78 | 1.00 | 5.87% |
| 7 | 0.169974 | 0.468781 | 0.063619 | 0.645618 | 10.84 | 1.00 | 7.43% |
| 8 | 0.121573 | 0.433174 | 0.055844 | 0.607459 | 10.84 | 1.00 | 6.98% |
| 9 | 0.113096 | 0.370850 | 0.054373 | 0.561656 | 10.86 | 0.91 | 7.39% |
| 10 | 0.083137 | 0.527120 | 0.046192 | 0.568013 | 10.88 | 1.00 | 5.73% |
| 11 | 0.143795 | 0.430081 | 0.057246 | 0.550353 | 10.75 | 0.90 | 6.74% |
| 12 | 0.087578 | 0.458651 | 0.040726 | 0.577834 | 10.94 | 1.00 | 7.08% |
| 13 | 0.112363 | 0.837996 | 0.060048 | 0.646300 | 10.92 | 0.89 | 7.43% |
| 14 | 0.152626 | 0.791763 | 0.063992 | 0.784499 | 10.75 | 1.00 | 7.24% |
| 15 | 0.096085 | 0.277036 | 0.054474 | 0.724021 | 11.17 | 0.96 | 6.60% |
| 16 | 0.152413 | 0.262144 | 0.069952 | 0.632313 | 11.02 | 1.00 | 7.35% |
| 17 | 0.080749 | 0.320576 | 0.043936 | 0.490511 | 10.91 | 1.00 | 5.90% |
| 18 | 0.111385 | 0.537718 | 0.055363 | 0.503665 | 10.89 | 0.94 | 5.25% |
| 19 | 0.190253 | 0.330034 | 0.083789 | 0.523940 | 11.14 | 1.00 | 8.29% |
| 20 | 0.080033 | 0.614957 | 0.042530 | 0.623505 | 10.94 | 1.00 | 5.52% |
| 21 | 0.107523 | 0.480291 | 0.058016 | 0.718650 | 11.08 | 0.96 | 7.08% |
| 22 | 0.136433 | 0.522900 | 0.071790 | 0.656845 | 10.91 | 1.00 | 7.37% |
| 23 | 0.102498 | 0.750286 | 0.057395 | 0.623250 | 10.75 | 1.00 | 5.66% |
| 24 | 0.093574 | 0.644644 | 0.054303 | 0.681420 | 10.78 | 1.00 | 5.96% |
| 25 | 0.083291 | 0.469535 | 0.044505 | 0.695727 | 10.95 | 1.00 | 5.09% |
| 26 | 0.087546 | 0.523036 | 0.047630 | 0.482964 | 10.83 | 1.00 | 4.54% |
| 27 | 0.120269 | 0.375248 | 0.057480 | 0.601136 | 10.78 | 0.95 | 6.70% |
| 28 | 0.085607 | 0.506550 | 0.052961 | 0.709872 | 10.94 | 1.00 | 5.58% |
| 29 | 0.165933 | 0.341772 | 0.070064 | 0.546702 | 11.03 | 1.00 | 9.03% |
| 30 | 0.123050 | 0.365789 | 0.060847 | 0.509688 | 10.69 | 1.00 | 5.21% |
| Best | 0.077429 | 0.262144 | 0.040726 | 0.482964 | 10.69 | 0.89 | 4.54% |
| Worst | 0.190253 | 0.837996 | 0.083789 | 0.784499 | 11.17 | 1.00 | 9.03% |
| Average | 0.111560 | 0.484115 | 0.055750 | 0.613799 | 10.89 | 0.98 | 6.45% |
| Median | 0.100196 | 0.469158 | 0.054919 | 0.611752 | 10.89 | 1.00 | 6.50% |
| Std.Dev. | 0.030396 | 0.149809 | 0.009659 | 0.078719 | 0.12 | 0.03 | 1.01% |

Table A.16 – MOPSO Statistics for Test Problem 4

| Run | GD | SP | iGD | DI | PT | DR | DD |
|---|---|---|---|---|---|---|---|
| 1 | 0.481645 | 0.021314 | 0.424287 | 0.958550 | 155.69 | 1.00 | 267.14% |
| 2 | 0.456468 | 0.003106 | 0.506032 | 0.987443 | 127.98 | 1.00 | 229.98% |
| 3 | 5.742791 | - | 0.640625 | - | 31.55 | 1.00 | 287.14% |
| 4 | 0.958778 | 0.008043 | 0.960764 | 0.985278 | 103.42 | 1.00 | 488.22% |
| 5 | 1.503558 | 0.033750 | 1.361384 | 0.971980 | 117.19 | 1.00 | 777.44% |
| 6 | 0.494248 | 0.000712 | 0.556254 | 0.996893 | 98.56 | 1.00 | 247.15% |
| 7 | 1.332746 | 0.000000 | 1.086562 | 0.999998 | 32.25 | 1.00 | 511.85% |
| 8 | 4.494310 | - | 0.517128 | - | 31.47 | 1.00 | 224.72% |
| 9 | 0.970560 | 0.002834 | 1.007550 | 0.992693 | 95.58 | 1.00 | 487.64% |
| 10 | 0.264654 | 0.012933 | 0.230664 | 0.916045 | 173.38 | 1.00 | 156.84% |
| 11 | 0.786604 | 0.001639 | 0.839792 | 0.996656 | 93.94 | 1.00 | 393.40% |
| 12 | 0.420726 | 0.002613 | 0.464196 | 0.985240 | 123.20 | 1.00 | 215.09% |
| 13 | 9.238348 | - | 0.987374 | - | 46.25 | 1.00 | 461.92% |
| 14 | 0.599670 | 0.025636 | 0.579751 | 0.973233 | 111.41 | 1.00 | 307.47% |
| 15 | 1.111959 | 0.010649 | 1.116301 | 0.989763 | 154.27 | 1.00 | 563.67% |
| 16 | 0.694341 | 0.024302 | 0.607770 | 0.958071 | 119.12 | 1.00 | 372.04% |
| 17 | 5.992425 | - | 0.665351 | - | 30.14 | 1.00 | 299.62% |
| 18 | 9.987351 | - | 1.061764 | - | 72.47 | 1.00 | 499.37% |
| 19 | 0.288018 | 0.007988 | 0.272961 | 0.941083 | 154.41 | 1.00 | 162.64% |
| 20 | 0.634665 | 0.007537 | 0.655979 | 0.987742 | 112.31 | 1.00 | 323.34% |
| 21 | 0.848864 | 0.020882 | 0.754167 | 0.958055 | 163.30 | 1.00 | 450.42% |
| 22 | 0.617276 | 0.004121 | 0.651827 | 0.989741 | 126.84 | 1.00 | 312.98% |
| 23 | 1.479449 | 0.002077 | 1.523901 | 0.997106 | 73.38 | 1.00 | 739.85% |
| 24 | 4.813213 | - | 0.548642 | - | 30.12 | 1.00 | 240.66% |
| 25 | 0.886001 | 0.001447 | 0.937118 | 0.996246 | 83.88 | 1.00 | 443.09% |
| 26 | 0.838099 | 0.020721 | 0.754027 | 0.958991 | 164.34 | 1.00 | 447.02% |
| 27 | 0.220456 | 0.018240 | 0.194534 | 0.892532 | 125.36 | 1.00 | 134.66% |
| 28 | 3.994941 | - | 0.467840 | - | 43.06 | 1.00 | 199.75% |
| 29 | 0.242813 | 0.020309 | 0.212391 | 0.945001 | 182.77 | 1.00 | 145.74% |
| 30 | 4.743996 | - | 0.541800 | - | 44.98 | 1.00 | 237.20% |
| | | | | | | | |
| Best | 0.220456 | 0.000000 | 0.194534 | 0.892532 | 30.12 | 1.00 | 134.66% |
| Worst | 9.987351 | 0.033750 | 1.523901 | 0.999998 | 182.77 | 1.00 | 777.44% |
| Average | 2.171299 | 0.011402 | 0.704291 | 0.971743 | 100.75 | 1.00 | 354.27% |
| Median | 0.867433 | 0.008016 | 0.646226 | 0.985259 | 107.42 | 1.00 | 310.23% |
| Std.Dev. | 2.686336 | 0.010020 | 0.329530 | 0.028448 | 48.31 | 0.00 | 165.39% |

Table A.17 – MOPSO-e Statistics for Test Problem 4

| Run | GD | SP | iGD | DI | PT | DR | DD |
|---|---|---|---|---|---|---|---|
| 1 | 0.200518 | 0.023516 | 0.176844 | 0.911695 | 163.09 | 1.00 | 124.09% |
| 2 | 0.159327 | 0.012629 | 0.142795 | 0.931542 | 206.28 | 1.00 | 102.48% |
| 3 | 1.147719 | 0.022342 | 1.030993 | 0.960006 | 126.67 | 1.00 | 600.12% |
| 4 | 0.579914 | 0.024471 | 0.527560 | 0.965997 | 156.09 | 1.00 | 314.64% |
| 5 | 0.307885 | 0.024079 | 0.267994 | 0.959962 | 174.20 | 1.00 | 178.66% |
| 6 | 0.705136 | 0.023529 | 0.630854 | 0.970080 | 140.27 | 1.00 | 379.73% |
| 7 | 0.249244 | 0.013633 | 0.212367 | 0.916017 | 180.98 | 1.00 | 147.27% |
| 8 | 0.053971 | 0.014828 | 0.052335 | 0.906021 | 210.20 | 1.00 | 40.00% |
| 9 | 0.133377 | 0.017049 | 0.135347 | 0.936885 | 194.84 | 1.00 | 84.65% |
| 10 | 0.070046 | 0.013570 | 0.066315 | 0.836726 | 115.03 | 1.00 | 50.54% |
| 11 | 0.263815 | 0.017999 | 0.230659 | 0.927488 | 169.53 | 1.00 | 156.68% |
| 12 | 0.232852 | 0.004853 | 0.256545 | 0.975157 | 138.33 | 1.00 | 126.27% |
| 13 | 1.747843 | - | 0.248041 | - | 29.14 | 1.00 | 87.39% |
| 14 | 0.386568 | 0.019410 | 0.344945 | 0.949657 | 233.36 | 1.00 | 220.80% |
| 15 | 0.144191 | 0.012356 | 0.126546 | 0.914301 | 145.69 | 1.00 | 92.92% |
| 16 | 0.489856 | 0.001266 | 0.547158 | 0.996621 | 133.91 | 1.00 | 245.01% |
| 17 | 0.681908 | 0.022075 | 0.607557 | 0.966283 | 229.41 | 1.00 | 368.10% |
| 18 | 0.566279 | 0.022705 | 0.506073 | 0.955347 | 216.62 | 1.00 | 310.69% |
| 19 | 0.484270 | 0.019075 | 0.424104 | 0.937172 | 140.94 | 1.00 | 267.76% |
| 20 | 0.811972 | 0.023704 | 0.711767 | 0.940624 | 182.75 | 1.00 | 430.44% |
| 21 | 0.764799 | 0.026860 | 0.690820 | 0.960321 | 167.27 | 1.00 | 411.07% |
| 22 | 0.163732 | 0.018709 | 0.159535 | 0.942217 | 201.73 | 1.00 | 109.54% |
| 23 | 0.797655 | 0.030837 | 0.712241 | 0.958053 | 168.31 | 1.00 | 425.82% |
| 24 | 0.261686 | 0.026426 | 0.230659 | 0.951769 | 182.19 | 1.00 | 156.15% |
| 25 | 0.396661 | 0.017418 | 0.344945 | 0.955079 | 202.59 | 1.00 | 223.45% |
| 26 | 0.324900 | 0.017584 | 0.287926 | 0.942668 | 153.42 | 1.00 | 188.54% |
| 27 | 0.370705 | 0.020917 | 0.325470 | 0.936102 | 169.14 | 1.00 | 211.24% |
| 28 | 0.406951 | 0.007017 | 0.425978 | 0.969511 | 104.92 | 1.00 | 211.47% |
| 29 | 0.668685 | 0.026242 | 0.587084 | 0.964061 | 144.59 | 1.00 | 359.62% |
| 30 | 0.218162 | 0.021065 | 0.194380 | 0.909242 | 121.66 | 1.00 | 134.05% |
| Best | 0.053971 | 0.001266 | 0.052335 | 0.836726 | 29.14 | 1.00 | 40.00% |
| Worst | 1.747843 | 0.030837 | 1.030993 | 0.996621 | 233.36 | 1.00 | 600.12% |
| Average | 0.459688 | 0.018833 | 0.373528 | 0.942986 | 163.44 | 1.00 | 225.31% |
| Median | 0.378637 | 0.019410 | 0.306698 | 0.949657 | 167.79 | 1.00 | 199.89% |
| Std.Dev. | 0.357437 | 0.006809 | 0.235557 | 0.030008 | 42.18 | 0.00 | 135.78% |

Table A.18 – MOPSO-p Statistics for Test Problem 4

| Run | GD | SP | iGD | DI | PT | DR | DD |
|---|---|---|---|---|---|---|---|
| 1 | 5.742727 | - | 0.640619 | - | 60.77 | 1.00 | 287.14% |
| 2 | 0.307341 | 0.012708 | 0.268017 | 0.931691 | 202.66 | 1.00 | 178.57% |
| 3 | 0.690037 | 0.011189 | 0.671452 | 0.975220 | 139.98 | 1.00 | 358.21% |
| 4 | 5.493065 | - | 0.615899 | - | 31.67 | 1.00 | 274.65% |
| 5 | 0.549373 | 0.020125 | 0.484850 | 0.945306 | 165.72 | 1.00 | 301.07% |
| 6 | 0.119670 | 0.018182 | 0.110814 | 0.900711 | 157.56 | 1.00 | 81.28% |
| 7 | 0.156332 | 0.015397 | 0.143399 | 0.908013 | 139.28 | 1.00 | 101.99% |
| 8 | 6.741466 | - | 0.739588 | - | 34.83 | 1.00 | 337.07% |
| 9 | 4.494321 | - | 0.517129 | - | 42.31 | 1.00 | 224.72% |
| 10 | 0.427755 | 0.024267 | 0.384352 | 0.959950 | 172.20 | 1.00 | 242.10% |
| 11 | 1.469973 | 0.000433 | 1.527412 | 0.999538 | 48.38 | 1.00 | 734.99% |
| 12 | 0.443690 | 0.024336 | 0.385273 | 0.966130 | 123.55 | 1.00 | 246.56% |
| 13 | 0.728099 | 0.024181 | 0.649180 | 0.950062 | 150.78 | 1.00 | 391.11% |
| 14 | 4.993676 | - | 0.566486 | - | 65.03 | 1.00 | 249.68% |
| 15 | 0.511897 | 0.004140 | 0.542538 | 0.985184 | 92.20 | 1.00 | 262.07% |
| 16 | 0.824874 | 0.025889 | 0.711951 | 0.950483 | 143.27 | 1.00 | 434.50% |
| 17 | 0.199741 | 0.000001 | 0.272141 | 0.999993 | 39.09 | 1.00 | 99.87% |
| 18 | 0.859656 | 0.019134 | 0.764532 | 0.971728 | 165.16 | 1.00 | 453.80% |
| 19 | 0.611707 | 0.004062 | 0.640833 | 0.983661 | 104.12 | 1.00 | 311.56% |
| 20 | 0.051862 | 0.010053 | 0.052518 | 0.904875 | 166.06 | 1.00 | 40.02% |
| 21 | 0.764087 | 0.032542 | 0.691595 | 0.962533 | 134.06 | 1.00 | 411.06% |
| 22 | 0.599227 | 0.000003 | 0.665319 | 0.999993 | 63.11 | 1.00 | 299.61% |
| 23 | 0.826054 | 0.023743 | 0.781457 | 0.974554 | 147.50 | 1.00 | 427.24% |
| 24 | 3.745266 | - | 0.443231 | - | 43.09 | 1.00 | 187.26% |
| 25 | 0.189755 | 0.008906 | 0.204200 | 0.949187 | 162.59 | 1.00 | 109.81% |
| 26 | 3.495574 | - | 0.418648 | - | 28.61 | 1.00 | 174.78% |
| 27 | 5.992424 | - | 0.665351 | - | 25.20 | 1.00 | 299.62% |
| 28 | 5.243365 | - | 0.591186 | - | 48.66 | 1.00 | 262.17% |
| 29 | 0.631947 | 0.005539 | 0.654623 | 0.980808 | 96.58 | 1.00 | 322.59% |
| 30 | 4.244625 | - | 0.492474 | - | 69.77 | 1.00 | 212.23% |
| | | | | | | | |
| Best | 0.051862 | 0.000001 | 0.052518 | 0.900711 | 25.20 | 1.00 | 40.02% |
| Worst | 6.741466 | 0.032542 | 1.527412 | 0.999993 | 202.66 | 1.00 | 734.99% |
| Average | 2.038320 | 0.014242 | 0.543236 | 0.959981 | 102.13 | 1.00 | 277.24% |
| Median | 0.746093 | 0.014053 | 0.578836 | 0.964332 | 100.35 | 1.00 | 268.41% |
| Std.Dev. | 2.235290 | 0.009928 | 0.275093 | 0.030430 | 54.98 | 0.00 | 138.64% |

Table A.19 – NSGA-II Statistics for Test Problem 4

| Run | GD | SP | iGD | DI | PT | DR | DD |
|---|---|---|---|---|---|---|---|
| 1 | 0.142195 | 0.010553 | 0.128305 | 0.763008 | 49.58 | 1.00 | 92.92% |
| 2 | 0.055377 | 0.007724 | 0.053013 | 0.680388 | 49.56 | 1.00 | 40.93% |
| 3 | 0.093357 | 0.010179 | 0.084999 | 0.741202 | 49.70 | 1.00 | 63.83% |
| 4 | 0.139535 | 0.010234 | 0.127350 | 0.765171 | 49.64 | 1.00 | 91.93% |
| 5 | 0.140100 | 0.010458 | 0.126799 | 0.768524 | 50.61 | 1.00 | 91.82% |
| 6 | 0.106839 | 0.009846 | 0.096208 | 0.753307 | 49.70 | 1.00 | 71.81% |
| 7 | 0.408050 | 0.013957 | 0.364796 | 0.859562 | 49.81 | 1.00 | 231.69% |
| 8 | 0.142619 | 0.010125 | 0.127035 | 0.781074 | 49.64 | 1.00 | 92.36% |
| 9 | 0.044234 | 0.008956 | 0.041542 | 0.648862 | 49.80 | 1.00 | 32.98% |
| 10 | 0.104901 | 0.010740 | 0.096497 | 0.751872 | 49.73 | 1.00 | 71.61% |
| 11 | 0.060033 | 0.008338 | 0.056592 | 0.662285 | 49.81 | 1.00 | 43.67% |
| 12 | 0.072341 | 0.009263 | 0.067191 | 0.727938 | 49.73 | 1.00 | 51.18% |
| 13 | 0.075246 | 0.008045 | 0.068950 | 0.691795 | 49.89 | 1.00 | 52.47% |
| 14 | 0.177429 | 0.010359 | 0.159749 | 0.792828 | 49.80 | 1.00 | 112.71% |
| 15 | 0.019990 | 0.007518 | 0.019858 | 0.526256 | 49.89 | 1.00 | 16.61% |
| 16 | 0.283409 | 0.010952 | 0.251649 | 0.839085 | 49.92 | 1.00 | 167.81% |
| 17 | 0.071820 | 0.009067 | 0.067045 | 0.698450 | 49.84 | 1.00 | 51.01% |
| 18 | 0.089504 | 0.010102 | 0.081159 | 0.726632 | 50.64 | 1.00 | 61.20% |
| 19 | 0.221223 | 0.012093 | 0.195157 | 0.814345 | 49.67 | 1.00 | 134.91% |
| 20 | 0.134098 | 0.011376 | 0.120633 | 0.784650 | 48.86 | 1.00 | 88.11% |
| 21 | 0.028019 | 0.008081 | 0.027125 | 0.624200 | 48.75 | 1.00 | 22.24% |
| 22 | 0.030149 | 0.008091 | 0.029435 | 0.611142 | 48.81 | 1.00 | 23.98% |
| 23 | 0.027777 | 0.007729 | 0.026618 | 0.596887 | 48.92 | 1.00 | 21.86% |
| 24 | 0.000857 | 0.007126 | 0.000581 | 0.342079 | 48.94 | 0.27 | 0.16% |
| 25 | 0.012853 | 0.008793 | 0.012555 | 0.537891 | 49.00 | 1.00 | 10.70% |
| 26 | 0.212091 | 0.010404 | 0.188839 | 0.820671 | 49.08 | 1.00 | 130.75% |
| 27 | 0.016507 | 0.006868 | 0.016183 | 0.541561 | 49.02 | 1.00 | 13.72% |
| 28 | 0.060922 | 0.007692 | 0.056965 | 0.671475 | 48.98 | 1.00 | 43.87% |
| 29 | 0.430774 | 0.014302 | 0.381279 | 0.868619 | 49.08 | 1.00 | 242.18% |
| 30 | 0.157706 | 0.011285 | 0.143671 | 0.798681 | 49.14 | 1.00 | 102.35% |
| Best | 0.000857 | 0.006868 | 0.000581 | 0.342079 | 48.75 | 0.27 | 0.16% |
| Worst | 0.430774 | 0.014302 | 0.381279 | 0.868619 | 50.64 | 1.00 | 242.18% |
| Average | 0.118665 | 0.009675 | 0.107259 | 0.706348 | 49.52 | 0.98 | 75.78% |
| Median | 0.091430 | 0.009974 | 0.083079 | 0.734570 | 49.66 | 1.00 | 62.52% |
| Std.Dev. | 0.106464 | 0.001849 | 0.094001 | 0.116360 | 0.49 | 0.13 | 59.68% |

Table A.20 – RSearch Statistics for Test Problem 4

| Run | GD | SP | iGD | DI | PT | DR | DD |
|---|---|---|---|---|---|---|---|
| 1 | 27.81641 | 6.86013 | 5.41368 | 0.90241 | 27.94 | 1.00 | 4650% |
| 2 | 27.08891 | 8.63632 | 4.35379 | 0.87524 | 28.33 | 1.00 | 5284% |
| 3 | 41.09725 | 26.16661 | 5.18614 | 0.93142 | 27.78 | 1.00 | 4687% |
| 4 | 28.26006 | 5.55673 | 5.48740 | 0.95131 | 27.73 | 1.00 | 4205% |
| 5 | 22.93513 | 3.82775 | 4.03029 | 0.88747 | 28.20 | 1.00 | 4529% |
| 6 | 26.66481 | 19.27086 | 3.74123 | 0.95958 | 28.00 | 1.00 | 4505% |
| 7 | 26.92655 | 19.73316 | 4.56968 | 0.99046 | 27.84 | 1.00 | 3928% |
| 8 | 30.15416 | 8.50726 | 4.37071 | 0.88980 | 28.00 | 1.00 | 4298% |
| 9 | 27.12436 | 12.03252 | 5.03564 | 1.01953 | 27.91 | 1.00 | 4918% |
| 10 | 35.49354 | 12.03842 | 4.16065 | 0.84232 | 27.72 | 1.00 | 4905% |
| 11 | 26.47104 | 6.83218 | 3.59258 | 0.83047 | 28.03 | 1.00 | 4725% |
| 12 | 27.31837 | 4.58852 | 5.29081 | 0.89383 | 28.02 | 1.00 | 4689% |
| 13 | 24.11343 | 13.83214 | 4.58413 | 0.98215 | 28.11 | 1.00 | 4645% |
| 14 | 26.44923 | 7.52195 | 4.51717 | 0.95418 | 28.22 | 1.00 | 4871% |
| 15 | 33.24148 | 38.39837 | 4.74041 | 1.04379 | 27.72 | 1.00 | 4320% |
| 16 | 27.99341 | 4.38187 | 5.22945 | 0.92224 | 27.94 | 1.00 | 5250% |
| 17 | 29.94525 | 7.76938 | 4.75425 | 0.96034 | 28.00 | 1.00 | 4306% |
| 18 | 33.22845 | 16.27287 | 4.11542 | 0.97395 | 28.00 | 1.00 | 5672% |
| 19 | 29.80387 | 11.86902 | 5.25730 | 0.93582 | 28.25 | 1.00 | 5092% |
| 20 | 28.00603 | 8.85518 | 4.32438 | 0.85722 | 29.09 | 1.00 | 4989% |
| 21 | 32.95688 | 7.27803 | 4.56279 | 0.96260 | 27.98 | 1.00 | 4985% |
| 22 | 25.77416 | 17.01384 | 4.83396 | 1.02239 | 27.84 | 1.00 | 4106% |
| 23 | 28.20282 | 4.00656 | 4.17710 | 0.91802 | 27.73 | 1.00 | 4385% |
| 24 | 24.13976 | 3.36262 | 4.14405 | 0.82944 | 28.00 | 1.00 | 4197% |
| 25 | 24.97295 | 15.00488 | 4.74596 | 0.95596 | 27.75 | 1.00 | 3920% |
| 26 | 27.16499 | 10.58283 | 5.36175 | 0.94505 | 28.23 | 1.00 | 5005% |
| 27 | 25.82101 | 12.86499 | 4.78848 | 1.02575 | 28.03 | 1.00 | 4475% |
| 28 | 24.87237 | 4.36126 | 5.07626 | 0.93279 | 28.03 | 1.00 | 4890% |
| 29 | 29.37025 | 4.62792 | 4.63318 | 0.88576 | 27.70 | 1.00 | 4221% |
| 30 | 25.67384 | 4.60207 | 5.42189 | 0.94167 | 27.61 | 1.00 | 4063% |
| | | | | | | | |
| Best | 22.93513 | 3.36262 | 3.59258 | 0.82944 | 27.61 | 1.00 | 3920% |
| Worst | 41.09725 | 38.39837 | 5.48740 | 1.04379 | 29.09 | 1.00 | 5672% |
| Average | 28.30269 | 10.88854 | 4.68335 | 0.93410 | 27.99 | 1.00 | 4624% |
| Median | 27.24168 | 8.57179 | 4.68679 | 0.93874 | 28.00 | 1.00 | 4647% |
| Std.Dev. | 3.79735 | 7.66925 | 0.51891 | 0.05727 | 0.28 | 0.00 | 430% |

Table A.21 – MOPSO Statistics for Test Problem 5

| Run | GD | SP | iGD | DI | PT | DR | DD |
|---|---|---|---|---|---|---|---|
| 1 | 0.000648 | 0.008139 | 0.000916 | 0.801498 | 8.00 | 0.58 | 0.63% |
| 2 | 0.000783 | 0.008010 | 0.001073 | 0.852982 | 7.61 | 0.51 | 0.87% |
| 3 | 0.000634 | 0.007248 | 0.000985 | 0.793525 | 8.03 | 0.52 | 0.74% |
| 4 | 0.000621 | 0.008049 | 0.001021 | 0.836404 | 7.34 | 0.49 | 0.71% |
| 5 | 0.000603 | 0.006933 | 0.001090 | 0.857994 | 7.50 | 0.52 | 0.51% |
| 6 | 0.000670 | 0.007125 | 0.000840 | 0.791170 | 7.80 | 0.47 | 0.89% |
| 7 | 0.000704 | 0.007127 | 0.000955 | 0.802665 | 7.62 | 0.51 | 0.68% |
| 8 | 0.000610 | 0.007631 | 0.001111 | 0.854924 | 7.69 | 0.46 | 0.53% |
| 9 | 0.000691 | 0.007639 | 0.001077 | 0.810143 | 8.11 | 0.58 | 0.58% |
| 10 | 0.000646 | 0.006676 | 0.000882 | 0.843974 | 7.75 | 0.60 | 0.79% |
| 11 | 0.000603 | 0.008320 | 0.001249 | 0.873136 | 7.11 | 0.57 | 1.13% |
| 12 | 0.000715 | 0.006777 | 0.001005 | 0.768252 | 7.94 | 0.62 | 0.63% |
| 13 | 0.000577 | 0.007424 | 0.000862 | 0.831625 | 7.69 | 0.48 | 0.80% |
| 14 | 0.000622 | 0.008041 | 0.001105 | 0.881882 | 7.75 | 0.58 | 0.58% |
| 15 | 0.000662 | 0.007213 | 0.000931 | 0.868528 | 7.73 | 0.53 | 0.48% |
| 16 | 0.000576 | 0.006250 | 0.000929 | 0.836470 | 7.62 | 0.49 | 1.16% |
| 17 | 0.000606 | 0.008401 | 0.000862 | 0.802307 | 6.98 | 0.48 | 0.81% |
| 18 | 0.000625 | 0.006357 | 0.001555 | 0.867036 | 7.47 | 0.44 | 0.77% |
| 19 | 0.000582 | 0.007623 | 0.000784 | 0.791774 | 7.64 | 0.52 | 0.61% |
| 20 | 0.000686 | 0.006666 | 0.001067 | 0.821779 | 7.72 | 0.54 | 0.74% |
| 21 | 0.000552 | 0.006400 | 0.001006 | 0.898569 | 7.20 | 0.50 | 0.51% |
| 22 | 0.000605 | 0.009609 | 0.001065 | 0.856853 | 7.77 | 0.44 | 0.85% |
| 23 | 0.000684 | 0.008285 | 0.000920 | 0.851709 | 8.20 | 0.57 | 0.46% |
| 24 | 0.000634 | 0.006702 | 0.000898 | 0.814840 | 7.84 | 0.53 | 0.80% |
| 25 | 0.000664 | 0.005992 | 0.000859 | 0.805822 | 8.02 | 0.52 | 0.55% |
| 26 | 0.000670 | 0.007087 | 0.001076 | 0.854232 | 7.89 | 0.52 | 0.55% |
| 27 | 0.000597 | 0.006014 | 0.000991 | 0.856323 | 8.34 | 0.52 | 0.71% |
| 28 | 0.000632 | 0.008663 | 0.000867 | 0.815920 | 7.64 | 0.55 | 0.47% |
| 29 | 0.000663 | 0.008003 | 0.001289 | 0.894638 | 7.92 | 0.60 | 1.00% |
| 30 | 0.000692 | 0.007449 | 0.001142 | 0.914299 | 8.00 | 0.60 | 0.53% |
| | | | | | | | |
| Best | 0.000552 | 0.005992 | 0.000784 | 0.768252 | 6.98 | 0.44 | 0.46% |
| Worst | 0.000783 | 0.009609 | 0.001555 | 0.914299 | 8.34 | 0.62 | 1.16% |
| Average | 0.000642 | 0.007395 | 0.001014 | 0.838376 | 7.73 | 0.53 | 0.70% |
| Median | 0.000634 | 0.007336 | 0.000998 | 0.840222 | 7.74 | 0.52 | 0.70% |
| Std.Dev. | 0.000049 | 0.000854 | 0.000158 | 0.036146 | 0.31 | 0.05 | 0.19% |

Table A.22 – MOPSO-e Statistics for Test Problem 5

| Run | GD | SP | iGD | DI | PT | DR | DD |
|---|---|---|---|---|---|---|---|
| 1 | 0.000654 | 0.007950 | 0.001121 | 0.836552 | 11.06 | 0.51 | 0.89% |
| 2 | 0.000568 | 0.008831 | 0.000845 | 0.789856 | 10.31 | 0.47 | 0.49% |
| 3 | 0.000619 | 0.008537 | 0.001144 | 0.879435 | 10.38 | 0.41 | 0.92% |
| 4 | 0.000584 | 0.007540 | 0.000933 | 0.869938 | 10.62 | 0.42 | 0.45% |
| 5 | 0.000623 | 0.007941 | 0.000899 | 0.824922 | 10.61 | 0.53 | 0.52% |
| 6 | 0.000583 | 0.007643 | 0.000978 | 0.820270 | 10.30 | 0.47 | 0.62% |
| 7 | 0.000606 | 0.006979 | 0.001023 | 0.855000 | 10.05 | 0.34 | 0.63% |
| 8 | 0.000563 | 0.007994 | 0.000897 | 0.787753 | 10.19 | 0.47 | 0.43% |
| 9 | 0.000620 | 0.007140 | 0.001162 | 0.888371 | 10.84 | 0.43 | 0.77% |
| 10 | 0.000553 | 0.009121 | 0.001053 | 0.917433 | 10.31 | 0.42 | 0.74% |
| 11 | 0.000611 | 0.008182 | 0.000962 | 0.836135 | 10.70 | 0.54 | 0.53% |
| 12 | 0.000594 | 0.007285 | 0.000947 | 0.812316 | 10.25 | 0.35 | 0.98% |
| 13 | 0.000560 | 0.006381 | 0.001013 | 0.827953 | 10.52 | 0.50 | 1.02% |
| 14 | 0.000545 | 0.007159 | 0.000921 | 0.908368 | 10.20 | 0.38 | 0.58% |
| 15 | 0.000583 | 0.008782 | 0.000824 | 0.860334 | 10.48 | 0.55 | 0.45% |
| 16 | 0.000551 | 0.007662 | 0.001061 | 0.898870 | 10.47 | 0.45 | 2.02% |
| 17 | 0.000515 | 0.007661 | 0.000891 | 0.861134 | 10.55 | 0.46 | 0.40% |
| 18 | 0.000561 | 0.007156 | 0.000841 | 0.818278 | 10.84 | 0.49 | 0.54% |
| 19 | 0.000515 | 0.006698 | 0.000955 | 0.863462 | 9.77 | 0.41 | 0.46% |
| 20 | 0.000595 | 0.006155 | 0.000806 | 0.782421 | 10.48 | 0.54 | 0.57% |
| 21 | 0.000595 | 0.008167 | 0.000917 | 0.807843 | 10.02 | 0.42 | 0.73% |
| 22 | 0.000592 | 0.007577 | 0.000903 | 0.822035 | 10.38 | 0.48 | 1.02% |
| 23 | 0.000592 | 0.007405 | 0.000949 | 0.883583 | 10.41 | 0.49 | 1.54% |
| 24 | 0.000500 | 0.007427 | 0.001004 | 0.844225 | 9.81 | 0.41 | 0.31% |
| 25 | 0.000604 | 0.007014 | 0.001045 | 0.856978 | 10.75 | 0.51 | 0.49% |
| 26 | 0.000540 | 0.007158 | 0.000873 | 0.828027 | 9.91 | 0.49 | 0.64% |
| 27 | 0.000545 | 0.006390 | 0.000982 | 0.846196 | 10.97 | 0.48 | 0.44% |
| 28 | 0.000660 | 0.007044 | 0.001099 | 0.843617 | 10.66 | 0.53 | 0.50% |
| 29 | 0.000568 | 0.009004 | 0.000934 | 0.859779 | 10.58 | 0.56 | 0.48% |
| 30 | 0.000574 | 0.007499 | 0.001328 | 0.917968 | 10.45 | 0.41 | 0.48% |
| Best | 0.000500 | 0.006155 | 0.000806 | 0.782421 | 9.77 | 0.34 | 0.31% |
| Worst | 0.000660 | 0.009121 | 0.001328 | 0.917968 | 11.06 | 0.56 | 2.02% |
| Average | 0.000579 | 0.007583 | 0.000977 | 0.848302 | 10.43 | 0.46 | 0.69% |
| Median | 0.000583 | 0.007520 | 0.000952 | 0.845211 | 10.46 | 0.47 | 0.56% |
| Std.Dev. | 0.000038 | 0.000766 | 0.000114 | 0.036817 | 0.32 | 0.06 | 0.36% |

Table A.23 – MOPSO-p Statistics for Test Problem 5

| Run | GD | SP | iGD | DI | PT | DR | DD |
|---|---|---|---|---|---|---|---|
| 1 | 0.000436 | 0.007428 | 0.000952 | 0.853137 | 8.31 | 0.08 | 3.25% |
| 2 | 0.000444 | 0.007603 | 0.000950 | 0.909236 | 8.66 | 0.08 | 0.08% |
| 3 | 0.000450 | 0.007471 | 0.001000 | 0.905202 | 7.94 | 0.09 | 0.06% |
| 4 | 0.000440 | 0.007350 | 0.001080 | 0.949674 | 8.59 | 0.07 | 0.08% |
| 5 | 0.000452 | 0.007066 | 0.001163 | 1.049799 | 8.22 | 0.17 | 0.83% |
| 6 | 0.000452 | 0.006923 | 0.000991 | 0.891141 | 7.86 | 0.07 | 0.08% |
| 7 | 0.000509 | 0.007118 | 0.000829 | 0.902687 | 8.44 | 0.09 | 0.09% |
| 8 | 0.000419 | 0.007352 | 0.000983 | 0.907223 | 7.80 | 0.11 | 1.24% |
| 9 | 0.000430 | 0.007306 | 0.000932 | 0.909473 | 8.14 | 0.06 | 0.09% |
| 10 | 0.000470 | 0.006165 | 0.000967 | 0.935394 | 8.05 | 0.12 | 0.06% |
| 11 | 0.000497 | 0.007979 | 0.000780 | 0.901495 | 8.33 | 0.10 | 0.09% |
| 12 | 0.000479 | 0.010678 | 0.000940 | 1.078403 | 8.02 | 0.12 | 1.21% |
| 13 | 0.000454 | 0.011045 | 0.001039 | 0.916609 | 8.33 | 0.06 | 2.14% |
| 14 | 0.000421 | 0.007987 | 0.000777 | 0.869743 | 7.56 | 0.10 | 1.35% |
| 15 | 0.000417 | 0.007558 | 0.001167 | 0.897875 | 8.11 | 0.11 | 1.20% |
| 16 | 0.000467 | 0.007495 | 0.000964 | 0.892601 | 7.97 | 0.09 | 1.51% |
| 17 | 0.000476 | 0.008750 | 0.000907 | 0.886036 | 8.61 | 0.09 | 0.07% |
| 18 | 0.000479 | 0.007544 | 0.001015 | 0.887691 | 8.30 | 0.09 | 0.07% |
| 19 | 0.000465 | 0.007096 | 0.001224 | 0.915583 | 7.97 | 0.10 | 1.35% |
| 20 | 0.000470 | 0.008072 | 0.000957 | 0.882402 | 8.11 | 0.09 | 1.47% |
| 21 | 0.000487 | 0.006365 | 0.000868 | 0.879103 | 8.08 | 0.09 | 0.08% |
| 22 | 0.000461 | 0.008995 | 0.000867 | 0.882499 | 7.92 | 0.06 | 0.10% |
| 23 | 0.000478 | 0.006745 | 0.001041 | 0.911149 | 8.11 | 0.10 | 2.81% |
| 24 | 0.000446 | 0.006967 | 0.000929 | 0.917833 | 7.75 | 0.09 | 1.49% |
| 25 | 0.000469 | 0.008669 | 0.000970 | 0.904561 | 8.05 | 0.11 | 1.22% |
| 26 | 0.000434 | 0.008133 | 0.000824 | 0.855294 | 8.22 | 0.10 | 0.10% |
| 27 | 0.000501 | 0.007283 | 0.001197 | 0.911904 | 7.41 | 0.13 | 0.08% |
| 28 | 0.000431 | 0.007577 | 0.001183 | 0.884767 | 8.23 | 0.12 | 0.08% |
| 29 | 0.000452 | 0.007472 | 0.000933 | 0.933419 | 8.16 | 0.09 | 0.07% |
| 30 | 0.000430 | 0.007916 | 0.001653 | 0.946261 | 8.08 | 0.15 | 0.06% |
| Best | 0.000417 | 0.006165 | 0.000777 | 0.853137 | 7.41 | 0.06 | 0.06% |
| Worst | 0.000509 | 0.011045 | 0.001653 | 1.078403 | 8.66 | 0.17 | 3.25% |
| Average | 0.000457 | 0.007737 | 0.001003 | 0.912273 | 8.11 | 0.10 | 0.75% |
| Median | 0.000453 | 0.007483 | 0.000965 | 0.904882 | 8.11 | 0.09 | 0.10% |
| Std.Dev. | 0.000025 | 0.001057 | 0.000171 | 0.047340 | 0.28 | 0.02 | 0.90% |

Table A.24 – NSGA-II Statistics for Test Problem 5

| Run | GD | SP | iGD | DI | PT | DR | DD |
|---|---|---|---|---|---|---|---|
| 1 | 0.000600 | 0.007793 | 0.008573 | 1.052413 | 12.89 | 0.36 | 0.64% |
| 2 | 0.000574 | 0.008004 | 0.015901 | 0.813471 | 12.94 | 0.44 | 0.54% |
| 3 | 0.002141 | 0.007855 | 0.009603 | 0.836777 | 12.92 | 0.56 | 3.11% |
| 4 | 0.000975 | 0.004627 | 0.002479 | 0.930929 | 12.92 | 0.52 | 1.38% |
| 5 | 0.000555 | 0.004664 | 0.015067 | 0.858909 | 12.92 | 0.43 | 0.68% |
| 6 | 0.001892 | 0.016009 | 0.006641 | 1.082372 | 12.88 | 0.52 | 1.01% |
| 7 | 0.000661 | 0.007069 | 0.014833 | 0.810255 | 12.97 | 0.46 | 0.88% |
| 8 | 0.000653 | 0.005383 | 0.002541 | 0.909033 | 12.89 | 0.43 | 0.54% |
| 9 | 0.002828 | 0.020943 | 0.006614 | 0.916175 | 13.08 | 0.61 | 2.88% |
| 10 | 0.001219 | 0.007007 | 0.001491 | 0.795775 | 13.33 | 0.49 | 0.85% |
| 11 | 0.000857 | 0.006787 | 0.001199 | 0.755457 | 12.95 | 0.49 | 1.13% |
| 12 | 0.001266 | 0.014470 | 0.007573 | 0.765833 | 13.02 | 0.49 | 2.39% |
| 13 | 0.000646 | 0.010258 | 0.001395 | 0.842426 | 12.92 | 0.42 | 0.39% |
| 14 | 0.000582 | 0.005610 | 0.001288 | 0.770430 | 12.86 | 0.52 | 0.97% |
| 15 | 0.001187 | 0.019305 | 0.007126 | 0.835692 | 12.94 | 0.43 | 1.87% |
| 16 | 0.000773 | 0.007499 | 0.003472 | 0.965358 | 12.80 | 0.52 | 0.71% |
| 17 | 0.000980 | 0.007810 | 0.001559 | 0.863980 | 12.94 | 0.48 | 0.89% |
| 18 | 0.000612 | 0.005793 | 0.001387 | 0.789580 | 12.84 | 0.44 | 0.38% |
| 19 | 0.000779 | 0.006099 | 0.002996 | 0.966976 | 12.89 | 0.37 | 0.73% |
| 20 | 0.000860 | 0.006394 | 0.003859 | 0.856182 | 12.95 | 0.36 | 1.59% |
| 21 | 0.000822 | 0.007552 | 0.003328 | 0.986369 | 12.92 | 0.51 | 0.55% |
| 22 | 0.000551 | 0.005862 | 0.001313 | 0.853712 | 12.89 | 0.42 | 0.38% |
| 23 | 0.000601 | 0.005157 | 0.002124 | 0.895987 | 12.80 | 0.47 | 0.40% |
| 24 | 0.003597 | 0.004841 | 0.011738 | 1.178969 | 12.59 | 0.43 | 2.45% |
| 25 | 0.000624 | 0.007660 | 0.001644 | 0.881694 | 12.91 | 0.48 | 0.48% |
| 26 | 0.000565 | 0.004824 | 0.002799 | 0.886554 | 12.81 | 0.35 | 0.88% |
| 27 | 0.000686 | 0.010235 | 0.002252 | 0.981478 | 12.88 | 0.48 | 0.57% |
| 28 | 0.000636 | 0.004550 | 0.001596 | 0.851665 | 12.92 | 0.51 | 0.69% |
| 29 | 0.000691 | 0.005928 | 0.001797 | 0.824927 | 12.84 | 0.56 | 0.45% |
| 30 | 0.001033 | 0.006342 | 0.002443 | 0.934468 | 12.83 | 0.42 | 1.04% |
| | | | | | | | |
| Best | 0.000551 | 0.004550 | 0.001199 | 0.755457 | 12.59 | 0.35 | 0.38% |
| Worst | 0.003597 | 0.020943 | 0.015901 | 1.178969 | 13.33 | 0.61 | 3.11% |
| Average | 0.001015 | 0.008078 | 0.004888 | 0.889795 | 12.91 | 0.47 | 1.05% |
| Median | 0.000732 | 0.006897 | 0.002670 | 0.861445 | 12.92 | 0.48 | 0.79% |
| Std.Dev. | 0.000713 | 0.004206 | 0.004501 | 0.097876 | 0.12 | 0.06 | 0.76% |

Table A.25 – RSearch Statistics for Test Problem 5

| Run | GD | SP | iGD | DI | PT | DR | DD |
|---|---|---|---|---|---|---|---|
| 1 | 0.002647 | 0.020833 | 0.002596 | 0.806842 | 10.27 | 0.84 | 1.51% |
| 2 | 0.001689 | 0.011831 | 0.002286 | 0.816946 | 10.11 | 0.88 | 1.52% |
| 3 | 0.001490 | 0.015685 | 0.001662 | 0.767432 | 10.19 | 0.78 | 0.98% |
| 4 | 0.001450 | 0.016090 | 0.001570 | 0.775681 | 10.34 | 0.79 | 1.67% |
| 5 | 0.002214 | 0.013705 | 0.001929 | 0.780575 | 10.36 | 0.84 | 1.41% |
| 6 | 0.001512 | 0.012962 | 0.001836 | 0.740485 | 10.11 | 0.82 | 1.31% |
| 7 | 0.002562 | 0.011069 | 0.001774 | 0.643145 | 10.14 | 0.86 | 2.08% |
| 8 | 0.001872 | 0.009822 | 0.002264 | 0.840445 | 10.25 | 0.87 | 1.45% |
| 9 | 0.001503 | 0.018002 | 0.001674 | 0.731783 | 10.14 | 0.72 | 1.10% |
| 10 | 0.001651 | 0.012105 | 0.001797 | 0.685167 | 10.28 | 0.78 | 1.67% |
| 11 | 0.001662 | 0.010708 | 0.001903 | 0.703373 | 10.33 | 0.65 | 1.53% |
| 12 | 0.001841 | 0.013054 | 0.001698 | 0.721992 | 10.41 | 0.78 | 2.13% |
| 13 | 0.001946 | 0.011548 | 0.002139 | 0.719748 | 10.33 | 0.81 | 1.73% |
| 14 | 0.001819 | 0.014440 | 0.001670 | 0.754907 | 10.25 | 0.79 | 1.78% |
| 15 | 0.002130 | 0.015265 | 0.001834 | 0.717250 | 10.28 | 0.82 | 2.12% |
| 16 | 0.001903 | 0.012324 | 0.001844 | 0.754166 | 10.48 | 0.85 | 1.78% |
| 17 | 0.001865 | 0.014111 | 0.001697 | 0.747310 | 10.27 | 0.86 | 1.33% |
| 18 | 0.001745 | 0.014080 | 0.001815 | 0.734037 | 10.20 | 0.68 | 1.17% |
| 19 | 0.001791 | 0.012875 | 0.001673 | 0.791445 | 10.19 | 0.79 | 1.28% |
| 20 | 0.001867 | 0.016562 | 0.001881 | 0.787901 | 10.31 | 0.74 | 1.88% |
| 21 | 0.002087 | 0.012683 | 0.002059 | 0.803357 | 10.47 | 0.87 | 1.85% |
| 22 | 0.002098 | 0.012840 | 0.001961 | 0.772081 | 10.44 | 0.88 | 1.86% |
| 23 | 0.002065 | 0.023972 | 0.002298 | 0.778536 | 10.17 | 0.91 | 1.51% |
| 24 | 0.001439 | 0.010929 | 0.001692 | 0.765053 | 10.47 | 0.91 | 1.52% |
| 25 | 0.001563 | 0.014036 | 0.002735 | 0.876383 | 10.14 | 0.71 | 1.29% |
| 26 | 0.002311 | 0.020388 | 0.002298 | 0.773147 | 10.14 | 0.83 | 1.35% |
| 27 | 0.001687 | 0.013527 | 0.001682 | 0.696971 | 10.22 | 0.86 | 1.88% |
| 28 | 0.001742 | 0.019445 | 0.002048 | 0.826064 | 10.19 | 0.86 | 1.27% |
| 29 | 0.002088 | 0.014620 | 0.002293 | 0.785521 | 10.23 | 0.86 | 1.36% |
| 30 | 0.002246 | 0.015597 | 0.002512 | 0.779137 | 10.23 | 0.84 | 1.78% |
| | | | | | | | |
| Best | 0.001439 | 0.009822 | 0.001570 | 0.643145 | 10.11 | 0.65 | 0.98% |
| Worst | 0.002647 | 0.023972 | 0.002735 | 0.876383 | 10.48 | 0.91 | 2.13% |
| Average | 0.001883 | 0.014504 | 0.001971 | 0.762563 | 10.26 | 0.82 | 1.57% |
| Median | 0.001853 | 0.013870 | 0.001863 | 0.769756 | 10.25 | 0.84 | 1.52% |
| Std.Dev. | 0.000314 | 0.003288 | 0.000311 | 0.048812 | 0.11 | 0.06 | 0.30% |

Table A.26 – MOPSO Statistics for Test Problem WE

| Run | GD | SP | iGD | DI | PT | DR | DD |
|---|---|---|---|---|---|---|---|
| 1 | 18.2310 | 186.5456 | 12.3269 | 0.7156 | 30.78 | 0.65 | 0.64% |
| 2 | 7.0538 | 111.9215 | 12.4261 | 0.6406 | 30.62 | 0.55 | 0.60% |
| 3 | 8.6337 | 111.0426 | 13.7407 | 0.6922 | 29.59 | 0.60 | 0.53% |
| 4 | 35.3728 | 221.2752 | 16.3974 | 0.7481 | 30.06 | 0.61 | 0.63% |
| 5 | 29.1594 | 140.1565 | 11.5878 | 0.7123 | 29.38 | 0.68 | 0.75% |
| 6 | 15.4099 | 152.1526 | 12.0026 | 0.6527 | 29.97 | 0.52 | 0.63% |
| 7 | 7.3886 | 111.3221 | 13.4003 | 0.7204 | 30.64 | 0.60 | 0.62% |
| 8 | 37.1579 | 224.5826 | 12.6445 | 0.6867 | 30.16 | 0.60 | 0.78% |
| 9 | 11.7969 | 127.5179 | 12.6393 | 0.6506 | 29.97 | 0.50 | 0.78% |
| 10 | 17.5616 | 169.0112 | 13.0079 | 0.6814 | 29.78 | 0.45 | 0.65% |
| 11 | 44.3133 | 394.3801 | 13.2896 | 0.7705 | 30.41 | 0.52 | 1.45% |
| 12 | 8.1152 | 111.1024 | 23.9754 | 0.6712 | 30.48 | 0.66 | 0.63% |
| 13 | 14.7662 | 147.3155 | 13.5971 | 0.6899 | 29.89 | 0.58 | 0.76% |
| 14 | 6.9915 | 100.7802 | 19.8232 | 0.6802 | 30.52 | 0.54 | 0.52% |
| 15 | 90.7161 | 120.5722 | 15.0577 | 0.7304 | 29.92 | 0.46 | 1.55% |
| 16 | 6.9188 | 111.0939 | 12.6719 | 0.6642 | 29.97 | 0.60 | 0.49% |
| 17 | 7.2926 | 109.7515 | 12.2297 | 0.6131 | 30.30 | 0.50 | 0.67% |
| 18 | 7.3722 | 124.6211 | 12.9050 | 0.6056 | 30.06 | 0.53 | 0.47% |
| 19 | 16.1446 | 192.6576 | 13.3021 | 0.6211 | 29.98 | 0.46 | 1.07% |
| 20 | 7.2924 | 101.4880 | 13.3855 | 0.6753 | 30.08 | 0.68 | 0.76% |
| 21 | 7.2506 | 159.7495 | 15.6219 | 0.6764 | 29.95 | 0.51 | 0.53% |
| 22 | 28.6595 | 263.7050 | 11.2246 | 0.6645 | 29.98 | 0.59 | 0.78% |
| 23 | 12.7494 | 147.6256 | 12.6798 | 0.7169 | 31.73 | 0.49 | 0.67% |
| 24 | 11.9306 | 166.6562 | 14.5742 | 0.6868 | 30.38 | 0.57 | 0.76% |
| 25 | 7.5388 | 112.2596 | 11.5405 | 0.5963 | 30.86 | 0.52 | 0.58% |
| 26 | 16.3990 | 141.4770 | 12.8846 | 0.6498 | 29.38 | 0.53 | 0.63% |
| 27 | 36.7784 | 259.9166 | 13.0523 | 0.7684 | 30.02 | 0.60 | 0.88% |
| 28 | 7.0266 | 100.0630 | 19.9056 | 0.6586 | 29.94 | 0.61 | 0.74% |
| 29 | 7.7764 | 109.5373 | 13.2199 | 0.6636 | 29.91 | 0.54 | 0.82% |
| 30 | 18.9105 | 183.8080 | 12.2136 | 0.6627 | 29.92 | 0.65 | 0.68% |
| Best | 6.9188 | 100.0630 | 11.2246 | 0.5963 | 29.38 | 0.45 | 0.47% |
| Worst | 90.7161 | 394.3801 | 23.9754 | 0.7705 | 31.73 | 0.68 | 1.55% |
| Average | 18.4236 | 157.1363 | 13.9109 | 0.6789 | 30.15 | 0.56 | 0.74% |
| Median | 12.3400 | 140.8167 | 13.0301 | 0.6759 | 30.00 | 0.56 | 0.67% |
| Std.Dev. | 17.4350 | 64.4669 | 2.7986 | 0.0435 | 0.47 | 0.07 | 0.24% |

Table A.27 – MOPSO-e Statistics for Test Problem WE

| Run | GD | SP | iGD | DI | PT | DR | DD |
|---|---|---|---|---|---|---|---|
| 1 | 8.4429 | 106.5936 | 15.2915 | 0.6196 | 49.66 | 0.53 | 0.58% |
| 2 | 17.1555 | 233.5672 | 14.3218 | 0.7299 | 49.66 | 0.58 | 0.85% |
| 3 | 7.7742 | 102.7671 | 14.8302 | 0.7019 | 48.22 | 0.49 | 0.52% |
| 4 | 7.4414 | 128.8448 | 12.9082 | 0.6806 | 47.59 | 0.54 | 0.52% |
| 5 | 6.9136 | 121.8188 | 10.9655 | 0.6187 | 48.59 | 0.45 | 0.42% |
| 6 | 7.2714 | 102.8204 | 11.6621 | 0.6579 | 48.22 | 0.48 | 0.49% |
| 7 | 7.0539 | 122.5828 | 13.1534 | 0.6901 | 48.80 | 0.48 | 0.58% |
| 8 | 6.6984 | 126.3815 | 15.3346 | 0.7003 | 49.73 | 0.56 | 0.47% |
| 9 | 10.6893 | 217.2968 | 17.5743 | 0.6740 | 50.06 | 0.47 | 0.45% |
| 10 | 7.3066 | 95.6797 | 13.8682 | 0.6481 | 48.06 | 0.51 | 0.51% |
| 11 | 7.4789 | 148.2208 | 12.8210 | 0.6358 | 48.94 | 0.50 | 0.54% |
| 12 | 12.4461 | 192.5232 | 15.4064 | 0.7039 | 48.02 | 0.45 | 0.55% |
| 13 | 7.3580 | 139.4359 | 13.3121 | 0.6785 | 49.59 | 0.57 | 0.70% |
| 14 | 7.9109 | 100.9801 | 11.6898 | 0.6131 | 49.02 | 0.60 | 0.48% |
| 15 | 6.3760 | 125.8202 | 13.1810 | 0.6977 | 48.41 | 0.53 | 0.54% |
| 16 | 7.7177 | 105.9764 | 12.6485 | 0.6670 | 49.14 | 0.50 | 0.47% |
| 17 | 8.3265 | 166.2635 | 16.8556 | 0.7145 | 48.98 | 0.48 | 0.41% |
| 18 | 7.1876 | 113.9967 | 12.0006 | 0.5761 | 48.09 | 0.48 | 0.47% |
| 19 | 7.1973 | 126.9017 | 13.0418 | 0.6557 | 48.39 | 0.57 | 0.57% |
| 20 | 6.8195 | 146.1781 | 13.2320 | 0.6933 | 49.31 | 0.57 | 0.55% |
| 21 | 7.2769 | 100.7543 | 21.1573 | 0.6009 | 48.03 | 0.51 | 0.46% |
| 22 | 7.2220 | 118.8680 | 17.5756 | 0.6719 | 51.09 | 0.48 | 0.53% |
| 23 | 6.5379 | 99.6060 | 13.3551 | 0.6991 | 48.58 | 0.63 | 0.65% |
| 24 | 7.0026 | 120.4070 | 12.6588 | 0.6506 | 48.14 | 0.45 | 0.53% |
| 25 | 7.3276 | 133.1476 | 19.7127 | 0.6473 | 49.91 | 0.52 | 0.57% |
| 26 | 13.2284 | 151.3900 | 15.3257 | 0.6882 | 48.62 | 0.50 | 0.42% |
| 27 | 7.6206 | 111.3504 | 15.5360 | 0.6632 | 49.44 | 0.53 | 0.51% |
| 28 | 7.4824 | 98.9361 | 15.8259 | 0.7416 | 48.53 | 0.57 | 0.56% |
| 29 | 7.6410 | 118.7877 | 13.1612 | 0.6481 | 49.27 | 0.49 | 0.45% |
| 30 | 6.3938 | 130.1378 | 14.2997 | 0.7108 | 49.00 | 0.51 | 0.49% |
| Best | 6.3760 | 95.6797 | 10.9655 | 0.5761 | 47.59 | 0.45 | 0.41% |
| Worst | 17.1555 | 233.5672 | 21.1573 | 0.7416 | 51.09 | 0.63 | 0.85% |
| Average | 8.1100 | 130.2678 | 14.4236 | 0.6693 | 48.90 | 0.52 | 0.53% |
| Median | 7.3428 | 122.2008 | 13.6116 | 0.6729 | 48.87 | 0.51 | 0.52% |
| Std.Dev. | 2.3281 | 33.8341 | 2.3538 | 0.0389 | 0.77 | 0.05 | 0.09% |

Table A.28 – MOPSO-p Statistics for Test Problem WE

| Run | GD | SP | iGD | DI | PT | DR | DD |
|---|---|---|---|---|---|---|---|
| 1 | 7.1355 | 105.3344 | 12.2378 | 0.6234 | 34.50 | 0.05 | 0.04% |
| 2 | 7.2795 | 149.9986 | 13.7580 | 0.6371 | 34.38 | 0.06 | 0.04% |
| 3 | 6.9506 | 109.9136 | 13.2666 | 0.6511 | 33.55 | 0.02 | 0.07% |
| 4 | 7.6914 | 111.5352 | 12.4051 | 0.5995 | 33.69 | 0.06 | 0.04% |
| 5 | 7.1500 | 108.9672 | 12.0693 | 0.6275 | 34.64 | 0.04 | 0.05% |
| 6 | 7.5880 | 129.1167 | 11.3795 | 0.6000 | 35.12 | 0.04 | 0.04% |
| 7 | 7.1758 | 108.7810 | 12.3964 | 0.6659 | 34.72 | 0.05 | 0.05% |
| 8 | 7.1071 | 130.6711 | 12.4785 | 0.6299 | 34.42 | 0.04 | 0.05% |
| 9 | 7.4857 | 113.1983 | 13.2050 | 0.6369 | 35.41 | 0.05 | 0.03% |
| 10 | 6.5175 | 133.5081 | 12.5381 | 0.6597 | 34.09 | 0.07 | 0.05% |
| 11 | 7.6255 | 111.6641 | 12.4125 | 0.6461 | 34.75 | 0.04 | 0.05% |
| 12 | 7.3394 | 142.9856 | 11.6823 | 0.6328 | 33.98 | 0.04 | 0.04% |
| 13 | 7.1324 | 138.3028 | 12.6622 | 0.6837 | 34.31 | 0.02 | 0.05% |
| 14 | 7.2339 | 165.0181 | 17.1550 | 0.6585 | 34.58 | 0.06 | 0.04% |
| 15 | 6.1824 | 106.0368 | 11.5024 | 0.6516 | 33.55 | 0.03 | 0.04% |
| 16 | 7.4328 | 125.0986 | 13.9126 | 0.7138 | 33.61 | 0.06 | 0.03% |
| 17 | 6.3597 | 146.7063 | 13.8544 | 0.7180 | 34.06 | 0.04 | 0.01% |
| 18 | 7.6534 | 108.2421 | 13.9771 | 0.6559 | 33.88 | 0.04 | 0.02% |
| 19 | 7.7574 | 125.3044 | 12.5954 | 0.6308 | 34.16 | 0.06 | 0.03% |
| 20 | 7.4353 | 114.7109 | 13.2371 | 0.6326 | 33.94 | 0.02 | 0.02% |
| 21 | 7.5022 | 104.1926 | 12.5581 | 0.5769 | 33.44 | 0.04 | 0.03% |
| 22 | 7.6154 | 106.1540 | 13.2670 | 0.6726 | 34.64 | 0.03 | 0.05% |
| 23 | 7.0132 | 145.0017 | 12.1978 | 0.7161 | 34.50 | 0.04 | 0.05% |
| 24 | 6.2701 | 116.7434 | 11.8454 | 0.6582 | 34.17 | 0.06 | 0.03% |
| 25 | 7.3515 | 130.1720 | 13.3900 | 0.6815 | 34.31 | 0.03 | 0.04% |
| 26 | 7.5013 | 101.9175 | 13.6867 | 0.6742 | 35.61 | 0.06 | 0.02% |
| 27 | 7.1142 | 136.1030 | 14.2321 | 0.6527 | 34.69 | 0.05 | 0.06% |
| 28 | 7.0333 | 103.4868 | 20.5988 | 0.7158 | 34.16 | 0.05 | 0.04% |
| 29 | 7.2204 | 127.6579 | 13.0983 | 0.6440 | 34.25 | 0.04 | 0.03% |
| 30 | 6.9766 | 99.2580 | 11.7827 | 0.6222 | 33.98 | 0.09 | 0.04% |
| | | | | | | | |
| Best | 6.1824 | 99.2580 | 11.3795 | 0.5769 | 33.44 | 0.02 | 0.01% |
| Worst | 7.7574 | 165.0181 | 20.5988 | 0.7180 | 35.61 | 0.09 | 0.07% |
| Average | 7.1944 | 121.8594 | 13.1794 | 0.6523 | 34.30 | 0.05 | 0.04% |
| Median | 7.2271 | 115.7272 | 12.6288 | 0.6513 | 34.28 | 0.04 | 0.04% |
| Std.Dev. | 0.4135 | 17.0946 | 1.7913 | 0.0348 | 0.52 | 0.02 | 0.01% |

Table A.29 – NSGA-II Statistics for Test Problem WE

| Run | GD | SP | iGD | DI | PT | DR | DD |
|---|---|---|---|---|---|---|---|
| 1 | 7.7642 | 114.6311 | 13.5211 | 0.5547 | 30.53 | 0.22 | 0.41% |
| 2 | 7.0882 | 120.2421 | 11.1599 | 0.4620 | 30.52 | 0.26 | 0.26% |
| 3 | 8.1545 | 106.9673 | 12.7089 | 0.5217 | 30.44 | 0.23 | 0.41% |
| 4 | 7.4050 | 130.8688 | 13.0759 | 0.5757 | 30.53 | 0.26 | 0.35% |
| 5 | 7.2694 | 95.2796 | 11.8348 | 0.5295 | 30.48 | 0.16 | 0.30% |
| 6 | 7.9749 | 119.0777 | 17.3357 | 0.5702 | 30.47 | 0.20 | 0.37% |
| 7 | 8.5064 | 116.3381 | 12.5131 | 0.4709 | 30.50 | 0.30 | 0.26% |
| 8 | 6.8897 | 88.1520 | 14.5863 | 0.5436 | 30.48 | 0.26 | 0.19% |
| 9 | 7.2112 | 117.3343 | 10.3253 | 0.4464 | 30.50 | 0.28 | 0.25% |
| 10 | 7.3519 | 103.2360 | 18.3762 | 0.5096 | 30.48 | 0.28 | 0.27% |
| 11 | 6.5596 | 111.2946 | 18.1387 | 0.5400 | 30.50 | 0.18 | 0.35% |
| 12 | 7.2169 | 109.2043 | 19.7364 | 0.5262 | 30.50 | 0.31 | 0.32% |
| 13 | 7.3147 | 149.4198 | 11.3941 | 0.4760 | 31.67 | 0.29 | 0.24% |
| 14 | 7.2415 | 128.5901 | 12.9994 | 0.5389 | 30.92 | 0.13 | 0.26% |
| 15 | 7.1460 | 106.0842 | 17.8520 | 0.5328 | 30.48 | 0.26 | 0.26% |
| 16 | 7.1746 | 128.8954 | 13.4135 | 0.5518 | 30.47 | 0.16 | 0.36% |
| 17 | 6.6339 | 111.3338 | 17.2887 | 0.5597 | 30.52 | 0.34 | 0.36% |
| 18 | 7.6756 | 97.1927 | 13.1993 | 0.4904 | 30.47 | 0.20 | 0.26% |
| 19 | 7.2584 | 112.7046 | 12.3873 | 0.5824 | 30.50 | 0.27 | 0.36% |
| 20 | 8.3593 | 113.7785 | 17.2189 | 0.6057 | 30.47 | 0.19 | 0.25% |
| 21 | 7.1405 | 142.8034 | 12.3802 | 0.5741 | 30.50 | 0.24 | 0.37% |
| 22 | 8.4289 | 121.0010 | 15.1293 | 0.5500 | 30.52 | 0.30 | 0.24% |
| 23 | 8.1156 | 145.2073 | 15.1600 | 0.5378 | 30.48 | 0.23 | 0.23% |
| 24 | 7.1990 | 140.5947 | 11.3989 | 0.5023 | 30.44 | 0.23 | 0.30% |
| 25 | 7.1497 | 125.8426 | 11.5056 | 0.4631 | 30.56 | 0.22 | 0.27% |
| 26 | 7.1951 | 116.3601 | 10.3419 | 0.5062 | 30.48 | 0.33 | 0.29% |
| 27 | 7.3197 | 116.2923 | 20.7053 | 0.5249 | 30.52 | 0.19 | 0.24% |
| 28 | 7.3902 | 106.2527 | 12.0994 | 0.5961 | 30.48 | 0.25 | 0.37% |
| 29 | 6.6918 | 119.3938 | 14.0511 | 0.5570 | 30.48 | 0.28 | 0.27% |
| 30 | 7.4836 | 138.2982 | 12.2723 | 0.5450 | 30.48 | 0.23 | 0.35% |
| | | | | | | | |
| Best | 6.5596 | 88.1520 | 10.3253 | 0.4464 | 30.44 | 0.13 | 0.19% |
| Worst | 8.5064 | 149.4198 | 20.7053 | 0.6057 | 31.67 | 0.34 | 0.41% |
| Average | 7.4103 | 118.4224 | 14.1370 | 0.5315 | 30.55 | 0.24 | 0.30% |
| Median | 7.2639 | 116.3491 | 13.1376 | 0.5383 | 30.49 | 0.24 | 0.28% |
| Std.Dev. | 0.5070 | 14.8875 | 2.8880 | 0.0407 | 0.23 | 0.05 | 0.06% |

Table A.30 – RSearch Statistics for Test Problem WE

| Run | GD | SP | iGD | DI | PT | DR | DD |
|---|---|---|---|---|---|---|---|
| 1 | 89.7749 | 344.9875 | 16.5807 | 0.7078 | 70.39 | 0.56 | 1.56% |
| 2 | 33.9324 | 236.5961 | 11.7979 | 0.6436 | 72.53 | 0.70 | 0.96% |
| 3 | 70.7056 | 463.5826 | 10.3940 | 0.7260 | 69.67 | 0.58 | 1.34% |
| 4 | 88.1455 | 523.6183 | 11.5987 | 0.7703 | 69.58 | 0.67 | 1.62% |
| 5 | 64.7533 | 339.3354 | 11.4818 | 0.7668 | 70.14 | 0.67 | 1.22% |
| 6 | 18.0990 | 109.0847 | 12.9545 | 0.5902 | 70.53 | 0.68 | 1.09% |
| 7 | 82.7598 | 192.5789 | 12.5454 | 0.7521 | 70.64 | 0.60 | 1.56% |
| 8 | 14.8597 | 130.3959 | 12.0734 | 0.5705 | 72.42 | 0.67 | 0.80% |
| 9 | 46.6987 | 120.1832 | 14.5205 | 0.6061 | 70.92 | 0.65 | 0.91% |
| 10 | 52.1422 | 398.0851 | 11.2413 | 0.7083 | 70.20 | 0.62 | 1.04% |
| 11 | 53.9811 | 474.6147 | 10.4036 | 0.7063 | 69.81 | 0.65 | 1.03% |
| 12 | 69.9526 | 210.1863 | 10.2538 | 0.6549 | 69.86 | 0.65 | 1.23% |
| 13 | 75.8575 | 385.1648 | 11.0049 | 0.7052 | 70.06 | 0.62 | 0.95% |
| 14 | 98.7506 | 233.8453 | 13.0719 | 0.7609 | 69.64 | 0.64 | 1.33% |
| 15 | 50.2109 | 326.7795 | 12.9956 | 0.6780 | 69.91 | 0.64 | 1.19% |
| 16 | 6.9375 | 113.2325 | 10.3365 | 0.5291 | 72.34 | 0.62 | 0.81% |
| 17 | 23.3324 | 164.4981 | 10.7534 | 0.5963 | 69.70 | 0.70 | 0.76% |
| 18 | 28.7657 | 299.9364 | 10.3551 | 0.6350 | 69.86 | 0.63 | 1.04% |
| 19 | 30.3937 | 233.0491 | 12.2503 | 0.6735 | 69.52 | 0.60 | 0.98% |
| 20 | 10.3884 | 92.1386 | 10.8926 | 0.5411 | 69.83 | 0.59 | 0.86% |
| 21 | 7.9715 | 96.4935 | 11.7450 | 0.6150 | 69.75 | 0.65 | 0.82% |
| 22 | 17.0460 | 145.4819 | 11.7950 | 0.6480 | 69.59 | 0.65 | 1.13% |
| 23 | 24.7158 | 95.4147 | 10.2778 | 0.5709 | 69.83 | 0.68 | 0.67% |
| 24 | 17.4303 | 192.5433 | 10.3410 | 0.5984 | 69.94 | 0.60 | 1.22% |
| 25 | 9.4053 | 166.9092 | 13.2882 | 0.5533 | 70.08 | 0.66 | 0.88% |
| 26 | 7.5204 | 86.8075 | 11.4416 | 0.5527 | 69.75 | 0.71 | 0.86% |
| 27 | 12.7983 | 91.0904 | 10.8332 | 0.5575 | 69.48 | 0.65 | 1.00% |
| 28 | 11.7948 | 120.8498 | 10.0406 | 0.5127 | 69.83 | 0.59 | 0.76% |
| 29 | 52.8417 | 242.2687 | 10.9383 | 0.6996 | 69.64 | 0.72 | 1.07% |
| 30 | 34.9252 | 251.2874 | 11.5609 | 0.6520 | 70.14 | 0.64 | 1.29% |
| | | | | | | | |
| Best | 6.9375 | 86.8075 | 10.0406 | 0.5127 | 69.48 | 0.56 | 0.67% |
| Worst | 98.7506 | 523.6183 | 16.5807 | 0.7703 | 72.53 | 0.72 | 1.62% |
| Average | 40.2297 | 229.3680 | 11.6589 | 0.6427 | 70.19 | 0.64 | 1.07% |
| Median | 32.1630 | 201.3826 | 11.4617 | 0.6458 | 69.86 | 0.65 | 1.04% |
| Std.Dev. | 28.7271 | 127.1075 | 1.4390 | 0.0765 | 0.83 | 0.04 | 0.25% |

Table A.31 – MOPSO Statistics for Test Problem WEF

| Run | GD | SP | iGD | PT | DR | DD |
|---|---|---|---|---|---|---|
| 1 | 55.7653 | 149.4049 | 6.5704 | 250.77 | 0.73 | 4.47% |
| 2 | 66.9412 | 180.7585 | 6.4597 | 245.31 | 0.72 | 4.21% |
| 3 | 52.8151 | 149.4106 | 7.0339 | 248.28 | 0.69 | 3.84% |
| 4 | 64.2009 | 157.4256 | 7.3325 | 243.97 | 0.70 | 4.19% |
| 5 | 71.7056 | 163.6307 | 7.0967 | 249.87 | 0.76 | 4.28% |
| 6 | 46.9160 | 161.2513 | 7.3392 | 250.70 | 0.75 | 4.02% |
| 7 | 77.7824 | 143.8649 | 8.1579 | 247.71 | 0.71 | 3.56% |
| 8 | 70.4577 | 165.9138 | 6.6433 | 245.30 | 0.78 | 4.44% |
| 9 | 75.2363 | 163.3368 | 7.6149 | 243.97 | 0.73 | 4.12% |
| 10 | 62.7133 | 153.1623 | 7.2324 | 248.70 | 0.61 | 3.87% |
| 11 | 62.2030 | 165.0859 | 7.3735 | 246.00 | 0.71 | 4.58% |
| 12 | 58.5537 | 161.1792 | 6.5795 | 245.98 | 0.75 | 3.84% |
| 13 | 46.7675 | 137.6455 | 7.4402 | 241.94 | 0.77 | 4.06% |
| 14 | 65.9046 | 157.7022 | 8.4527 | 245.22 | 0.72 | 3.44% |
| 15 | 55.2873 | 165.1605 | 7.2025 | 243.18 | 0.73 | 3.96% |
| 16 | 67.8646 | 156.6726 | 6.8692 | 245.48 | 0.77 | 4.32% |
| 17 | 63.0782 | 160.6449 | 7.2296 | 249.65 | 0.75 | 3.69% |
| 18 | 64.9710 | 161.6736 | 6.8255 | 249.03 | 0.75 | 3.82% |
| 19 | 75.4574 | 146.0203 | 7.0195 | 244.56 | 0.80 | 4.13% |
| 20 | 58.9599 | 138.4230 | 6.7378 | 244.66 | 0.72 | 3.58% |
| 21 | 41.9482 | 142.8947 | 6.4514 | 245.88 | 0.69 | 3.47% |
| 22 | 71.6861 | 150.4207 | 7.5003 | 246.94 | 0.69 | 3.83% |
| 23 | 69.5040 | 159.8994 | 6.6027 | 248.36 | 0.78 | 4.40% |
| 24 | 45.9884 | 165.7113 | 7.9878 | 243.56 | 0.70 | 4.82% |
| 25 | 72.6104 | 154.7743 | 6.8727 | 247.56 | 0.73 | 5.07% |
| 26 | 69.8188 | 148.6868 | 6.9298 | 244.52 | 0.78 | 5.08% |
| 27 | 71.3506 | 171.3895 | 6.7194 | 244.69 | 0.73 | 5.18% |
| 28 | 72.6539 | 145.7298 | 6.6483 | 249.88 | 0.71 | 4.17% |
| 29 | 58.6345 | 177.6165 | 6.2869 | 247.42 | 0.68 | 3.58% |
| 30 | 70.0925 | 177.5682 | 6.9773 | 242.75 | 0.77 | 4.48% |
| | | | | | | |
| Best | 41.9482 | 137.6455 | 6.2869 | 241.94 | 0.61 | 3.44% |
| Worst | 77.7824 | 180.7585 | 8.4527 | 250.77 | 0.80 | 5.18% |
| Average | 63.5956 | 157.7686 | 7.0729 | 246.39 | 0.73 | 4.15% |
| Median | 65.4378 | 158.8008 | 6.9984 | 245.93 | 0.73 | 4.13% |
| Std.Dev. | 9.6368 | 11.2024 | 0.5153 | 2.48 | 0.04 | 0.47% |

Table A.32 – MOPSO-e Statistics for Test Problem WEF

| Run | GD | SP | iGD | PT | DR | DD |
|---|---|---|---|---|---|---|
| 1 | 22.6313 | 354.6647 | 6.2705 | 409.59 | 0.59 | 2.77% |
| 2 | 67.6015 | 163.1476 | 6.3561 | 417.20 | 0.68 | 3.08% |
| 3 | 14.9944 | 226.0284 | 7.0067 | 413.20 | 0.59 | 2.53% |
| 4 | 6.8309 | 142.2131 | 7.1175 | 412.62 | 0.59 | 2.67% |
| 5 | 58.6747 | 138.0922 | 7.5343 | 417.16 | 0.61 | 3.10% |
| 6 | 53.0835 | 184.0468 | 7.2875 | 414.04 | 0.63 | 2.75% |
| 7 | 62.9529 | 240.9447 | 10.3075 | 408.69 | 0.66 | 2.88% |
| 8 | 7.2223 | 161.4415 | 6.8365 | 417.69 | 0.68 | 2.76% |
| 9 | 6.9943 | 133.6138 | 7.9048 | 414.86 | 0.61 | 2.85% |
| 10 | 6.9926 | 167.5319 | 6.8093 | 413.72 | 0.61 | 3.01% |
| 11 | 7.0039 | 118.3075 | 6.6531 | 404.97 | 0.59 | 2.84% |
| 12 | 6.7312 | 131.7514 | 6.7335 | 416.08 | 0.53 | 2.80% |
| 13 | 6.9661 | 121.5232 | 6.5541 | 415.14 | 0.69 | 2.69% |
| 14 | 21.1200 | 480.5696 | 11.1745 | 420.88 | 0.66 | 2.64% |
| 15 | 57.5798 | 158.0311 | 19.7866 | 420.50 | 0.61 | 3.31% |
| 16 | 8.3810 | 152.2665 | 6.4413 | 413.34 | 0.63 | 3.16% |
| 17 | 8.1317 | 246.3525 | 9.1635 | 414.86 | 0.61 | 3.31% |
| 18 | 17.8137 | 175.2042 | 7.0160 | 413.00 | 0.59 | 3.55% |
| 19 | 6.6693 | 142.7680 | 8.1718 | 407.95 | 0.66 | 2.78% |
| 20 | 7.1165 | 118.1548 | 6.3024 | 413.28 | 0.57 | 2.72% |
| 21 | 14.0844 | 174.8932 | 6.7415 | 424.94 | 0.63 | 3.02% |
| 22 | 28.2300 | 180.7583 | 5.8758 | 424.62 | 0.59 | 2.65% |
| 23 | 8.8785 | 133.8557 | 7.4254 | 409.50 | 0.66 | 2.84% |
| 24 | 8.0003 | 160.1819 | 5.9837 | 409.30 | 0.66 | 2.91% |
| 25 | 6.7824 | 135.6152 | 6.4111 | 412.45 | 0.74 | 2.78% |
| 26 | 7.9000 | 111.7221 | 6.7679 | 416.58 | 0.58 | 2.83% |
| 27 | 8.1565 | 118.4616 | 7.1399 | 405.28 | 0.57 | 2.74% |
| 28 | 22.4271 | 477.2157 | 9.4167 | 405.94 | 0.63 | 3.18% |
| 29 | 21.8396 | 374.7282 | 6.6465 | 412.56 | 0.70 | 2.76% |
| 30 | 32.1578 | 197.6610 | 6.7499 | 416.81 | 0.63 | 3.00% |
| | | | | | | |
| Best | 6.6693 | 111.7221 | 5.8758 | 404.97 | 0.53 | 2.53% |
| Worst | 67.6015 | 480.5696 | 19.7866 | 424.94 | 0.74 | 3.55% |
| Average | 20.4649 | 194.0582 | 7.6862 | 413.89 | 0.63 | 2.90% |
| Median | 8.6298 | 160.8117 | 6.8229 | 413.53 | 0.62 | 2.84% |
| Std.Dev. | 19.4255 | 99.4584 | 2.6009 | 5.04 | 0.05 | 0.23% |

Table A.33 – MOPSO-p Statistics for Test Problem WEF

| Run | GD | SP | iGD | PT | DR | DD |
|---|---|---|---|---|---|---|
| 1 | 6.5471 | 132.2945 | 7.1438 | 251.79 | 0.00 | 0.00% |
| 2 | 6.8864 | 132.5383 | 6.4450 | 253.10 | 0.00 | 0.00% |
| 3 | 7.0857 | 133.1920 | 6.6706 | 250.26 | 0.00 | 0.00% |
| 4 | 6.3011 | 133.0357 | 6.5864 | 248.09 | 0.01 | 0.00% |
| 5 | 6.6347 | 125.6099 | 6.6342 | 253.24 | 0.01 | 0.03% |
| 6 | 6.7962 | 150.3922 | 6.8541 | 253.90 | 0.00 | 0.00% |
| 7 | 6.4142 | 163.4627 | 6.8826 | 251.64 | 0.00 | 0.00% |
| 8 | 6.7650 | 159.6126 | 6.1568 | 248.71 | 0.00 | 0.00% |
| 9 | 6.5479 | 135.6558 | 6.2867 | 250.41 | 0.00 | 0.00% |
| 10 | 6.1899 | 125.0756 | 6.6458 | 255.24 | 0.00 | 0.00% |
| 11 | 6.4133 | 131.3210 | 7.4953 | 254.63 | 0.00 | 0.00% |
| 12 | 6.5477 | 140.1083 | 7.4190 | 254.46 | 0.00 | 0.00% |
| 13 | 6.7346 | 124.2391 | 7.2801 | 255.22 | 0.00 | 0.00% |
| 14 | 6.3060 | 149.6036 | 6.2280 | 248.60 | 0.00 | 0.00% |
| 15 | 6.7622 | 127.3428 | 6.6549 | 256.41 | 0.00 | 0.00% |
| 16 | 6.4503 | 182.9563 | 6.5287 | 255.55 | 0.00 | 0.00% |
| 17 | 6.3307 | 140.5934 | 6.1073 | 245.04 | 0.00 | 0.00% |
| 18 | 6.5186 | 137.6553 | 6.5158 | 249.15 | 0.00 | 0.00% |
| 19 | 6.8532 | 143.1536 | 6.5615 | 251.95 | 0.00 | 0.00% |
| 20 | 6.5918 | 114.5386 | 7.4136 | 253.94 | 0.00 | 0.00% |
| 21 | 6.4002 | 145.3534 | 6.4588 | 251.54 | 0.01 | 0.04% |
| 22 | 6.9596 | 144.2100 | 6.7155 | 261.48 | 0.00 | 0.00% |
| 23 | 6.3702 | 130.7137 | 6.0629 | 245.84 | 0.00 | 0.00% |
| 24 | 6.7641 | 115.4293 | 6.4280 | 246.76 | 0.00 | 0.00% |
| 25 | 6.6043 | 128.9635 | 7.1092 | 258.55 | 0.00 | 0.00% |
| 26 | 6.2538 | 138.7842 | 6.9897 | 245.37 | 0.01 | 0.04% |
| 27 | 6.8255 | 150.8803 | 7.0139 | 258.48 | 0.00 | 0.00% |
| 28 | 6.8505 | 139.3475 | 6.5307 | 258.52 | 0.01 | 0.02% |
| 29 | 6.4197 | 122.3462 | 6.4887 | 257.45 | 0.00 | 0.00% |
| 30 | 6.1041 | 157.3934 | 5.9099 | 246.16 | 0.00 | 0.00% |
| | | | | | | |
| Best | 6.1041 | 114.5386 | 5.9099 | 245.04 | 0.00 | 0.00% |
| Worst | 7.0857 | 182.9563 | 7.4953 | 261.48 | 0.01 | 0.04% |
| Average | 6.5743 | 138.5268 | 6.6739 | 252.38 | 0.00 | 0.00% |
| Median | 6.5478 | 136.6555 | 6.6103 | 252.53 | 0.00 | 0.00% |
| Std.Dev. | 0.2462 | 14.7563 | 0.4163 | 4.40 | 0.00 | 0.01% |

Table A.34 – NSGA-II Statistics for Test Problem WEF

| Run | GD | SP | iGD | PT | DR | DD |
|---|---|---|---|---|---|---|
| 1 | 5.8098 | 140.9486 | 5.8112 | 118.18 | 0.22 | 2.61% |
| 2 | 6.4453 | 123.4947 | 7.5711 | 117.03 | 0.18 | 2.08% |
| 3 | 6.8038 | 127.5345 | 6.3153 | 117.11 | 0.26 | 2.81% |
| 4 | 6.7407 | 140.7673 | 6.9159 | 116.55 | 0.26 | 2.47% |
| 5 | 7.2058 | 172.7231 | 5.8923 | 116.75 | 0.22 | 2.60% |
| 6 | 6.2803 | 149.1974 | 5.8890 | 117.11 | 0.22 | 2.65% |
| 7 | 7.6826 | 140.1385 | 6.0049 | 116.76 | 0.18 | 2.52% |
| 8 | 6.5215 | 144.9511 | 6.0339 | 116.94 | 0.16 | 3.26% |
| 9 | 6.2307 | 132.0277 | 6.2823 | 117.53 | 0.22 | 2.46% |
| 10 | 6.5535 | 144.2472 | 6.3066 | 117.01 | 0.22 | 2.31% |
| 11 | 6.6396 | 156.0817 | 5.6551 | 115.94 | 0.20 | 2.22% |
| 12 | 6.7809 | 143.0016 | 6.4331 | 117.23 | 0.26 | 2.21% |
| 13 | 6.5922 | 136.7915 | 6.5590 | 116.84 | 0.22 | 2.16% |
| 14 | 6.5810 | 148.7784 | 5.7869 | 117.36 | 0.22 | 2.23% |
| 15 | 6.7341 | 131.7921 | 6.9128 | 116.87 | 0.18 | 1.59% |
| 16 | 6.6159 | 156.0986 | 6.2742 | 116.69 | 0.18 | 2.48% |
| 17 | 6.4127 | 157.3761 | 6.3036 | 116.76 | 0.24 | 1.76% |
| 18 | 5.7930 | 135.0908 | 5.7855 | 116.44 | 0.18 | 2.04% |
| 19 | 6.8430 | 154.3097 | 6.0534 | 117.95 | 0.24 | 2.82% |
| 20 | 6.8079 | 152.8150 | 6.6470 | 117.59 | 0.15 | 1.91% |
| 21 | 6.0715 | 154.4402 | 6.2816 | 116.62 | 0.14 | 1.85% |
| 22 | 6.6728 | 142.0066 | 6.3674 | 116.06 | 0.19 | 2.76% |
| 23 | 6.7144 | 127.4792 | 6.4420 | 116.53 | 0.24 | 2.11% |
| 24 | 6.2833 | 140.6919 | 6.2650 | 115.90 | 0.26 | 2.59% |
| 25 | 6.8816 | 148.0917 | 5.9992 | 116.81 | 0.21 | 1.58% |
| 26 | 8.5683 | 179.7600 | 7.0232 | 116.89 | 0.24 | 2.01% |
| 27 | 7.0917 | 144.2079 | 6.1158 | 116.56 | 0.23 | 2.73% |
| 28 | 6.0852 | 136.0079 | 6.1568 | 116.95 | 0.16 | 1.89% |
| 29 | 6.9329 | 134.0451 | 7.4623 | 117.73 | 0.22 | 1.90% |
| 30 | 6.7130 | 135.0796 | 5.5399 | 116.16 | 0.20 | 2.26% |
| | | | | | | |
| Best | 5.7930 | 123.4947 | 5.5399 | 115.90 | 0.14 | 1.58% |
| Worst | 8.5683 | 179.7600 | 7.5711 | 118.18 | 0.26 | 3.26% |
| Average | 6.6696 | 144.3325 | 6.3029 | 116.90 | 0.21 | 2.30% |
| Median | 6.6562 | 142.5041 | 6.2779 | 116.86 | 0.22 | 2.25% |
| Std.Dev. | 0.5309 | 12.6132 | 0.4891 | 0.55 | 0.03 | 0.40% |

Table A.35 – RSearch Statistics for Test Problem WEF

| Run | GD | SP | iGD | PT | DR | DD |
|---|---|---|---|---|---|---|
| 1 | 60.2684 | 152.7515 | 7.4530 | 127.98 | 0.49 | 4.44% |
| 2 | 59.2720 | 142.3656 | 6.9639 | 125.25 | 0.55 | 4.33% |
| 3 | 51.8285 | 152.2419 | 7.8194 | 123.50 | 0.50 | 3.71% |
| 4 | 54.6509 | 157.4796 | 6.9790 | 124.42 | 0.51 | 3.76% |
| 5 | 43.8382 | 137.2114 | 6.0449 | 123.45 | 0.54 | 3.50% |
| 6 | 64.6985 | 157.7168 | 7.9582 | 124.53 | 0.50 | 4.57% |
| 7 | 53.7217 | 159.9727 | 7.1833 | 123.19 | 0.55 | 3.75% |
| 8 | 62.7086 | 151.1952 | 7.2279 | 126.03 | 0.59 | 4.19% |
| 9 | 56.9941 | 172.6779 | 8.3579 | 126.14 | 0.61 | 3.65% |
| 10 | 49.0192 | 152.9592 | 6.9109 | 125.04 | 0.51 | 4.04% |
| 11 | 47.1403 | 161.8171 | 7.4940 | 124.75 | 0.53 | 3.65% |
| 12 | 46.2469 | 132.6734 | 6.9417 | 122.95 | 0.57 | 4.00% |
| 13 | 51.7419 | 137.8265 | 7.5095 | 123.59 | 0.51 | 4.05% |
| 14 | 44.9584 | 165.2856 | 7.0526 | 123.23 | 0.57 | 4.44% |
| 15 | 42.4903 | 135.4737 | 7.5713 | 123.03 | 0.55 | 3.65% |
| 16 | 57.7998 | 137.9876 | 7.8120 | 123.42 | 0.50 | 4.74% |
| 17 | 47.5775 | 159.2957 | 7.9618 | 124.78 | 0.48 | 4.40% |
| 18 | 47.6860 | 141.6621 | 7.0895 | 123.29 | 0.47 | 3.93% |
| 19 | 54.1649 | 140.7957 | 8.0374 | 123.55 | 0.55 | 3.72% |
| 20 | 53.8305 | 166.6070 | 6.8728 | 121.75 | 0.61 | 3.85% |
| 21 | 66.0245 | 149.2966 | 7.3577 | 123.72 | 0.54 | 4.01% |
| 22 | 50.0448 | 137.0857 | 6.3666 | 123.48 | 0.52 | 4.16% |
| 23 | 51.0956 | 158.6662 | 6.3120 | 123.18 | 0.56 | 3.72% |
| 24 | 60.6558 | 151.8665 | 7.2730 | 122.28 | 0.54 | 4.03% |
| 25 | 56.1343 | 141.6856 | 7.0534 | 124.28 | 0.58 | 4.02% |
| 26 | 49.0594 | 155.1272 | 7.4836 | 123.78 | 0.58 | 4.38% |
| 27 | 52.6329 | 161.3529 | 8.0400 | 124.39 | 0.53 | 4.17% |
| 28 | 61.9857 | 164.0641 | 7.2482 | 122.64 | 0.44 | 3.75% |
| 29 | 50.1105 | 160.0495 | 6.6249 | 124.38 | 0.50 | 4.01% |
| 30 | 56.8406 | 142.0093 | 6.8951 | 124.18 | 0.53 | 3.31% |
| | | | | | | |
| Best | 42.4903 | 132.6734 | 6.0449 | 121.75 | 0.44 | 3.31% |
| Worst | 66.0245 | 172.6779 | 8.3579 | 127.98 | 0.61 | 4.74% |
| Average | 53.5074 | 151.2400 | 7.2632 | 124.01 | 0.53 | 4.00% |
| Median | 53.1773 | 152.4967 | 7.2381 | 123.66 | 0.54 | 4.01% |
| Std.Dev. | 6.2842 | 10.9134 | 0.5477 | 1.25 | 0.04 | 0.34% |

Table A.36 – MOPSO Statistics for Test Problem WEFR

| Run | GD | SP | iGD | PT | DR | DD |
|---|---|---|---|---|---|---|
| 1 | 16.1616 | 157.0574 | 3.0201 | 1405.13 | 0.72 | 5.70% |
| 2 | 15.4363 | 157.3252 | 3.6053 | 1390.76 | 0.70 | 4.76% |
| 3 | 14.0537 | 158.1183 | 2.9593 | 1398.09 | 0.67 | 5.45% |
| 4 | 14.6063 | 171.6088 | 3.5157 | 1404.09 | 0.66 | 6.09% |
| 5 | 12.8292 | 159.3614 | 3.0856 | 1389.91 | 0.66 | 5.80% |
| 6 | 15.1823 | 167.4518 | 2.8737 | 1419.63 | 0.69 | 5.01% |
| 7 | 13.3343 | 157.7490 | 2.8263 | 1403.84 | 0.64 | 5.37% |
| 8 | 13.3069 | 175.9165 | 2.9532 | 1412.32 | 0.62 | 4.55% |
| 9 | 15.2152 | 149.2104 | 3.1543 | 1395.55 | 0.71 | 5.54% |
| 10 | 16.9035 | 150.0096 | 3.3400 | 1411.71 | 0.72 | 5.97% |
| 11 | 14.8310 | 152.4279 | 3.3642 | 1254.00 | 0.71 | 6.22% |
| 12 | 16.2445 | 157.3535 | 2.9409 | 1270.03 | 0.65 | 5.54% |
| 13 | 14.8417 | 147.6817 | 2.9138 | 1285.16 | 0.69 | 5.51% |
| 14 | 14.2823 | 165.0206 | 2.9670 | 1274.52 | 0.70 | 5.31% |
| 15 | 16.8254 | 157.0966 | 3.0055 | 1280.86 | 0.75 | 6.56% |
| 16 | 16.9262 | 161.6320 | 2.8321 | 1273.35 | 0.77 | 5.30% |
| 17 | 15.5882 | 169.3300 | 2.9452 | 1258.59 | 0.65 | 5.45% |
| 18 | 16.1894 | 150.6199 | 3.2445 | 1263.49 | 0.63 | 5.57% |
| 19 | 15.8507 | 150.4828 | 3.4850 | 1259.16 | 0.67 | 5.59% |
| 20 | 14.8961 | 145.2007 | 2.9599 | 1258.22 | 0.66 | 4.93% |
| 21 | 13.4068 | 144.1521 | 2.7241 | 1287.69 | 0.59 | 4.96% |
| 22 | 15.7379 | 164.1057 | 3.0070 | 1267.70 | 0.62 | 5.50% |
| 23 | 17.0187 | 172.0657 | 2.9542 | 1281.88 | 0.70 | 5.24% |
| 24 | 15.2127 | 152.6937 | 3.4855 | 1281.22 | 0.73 | 5.18% |
| 25 | 14.2539 | 169.7852 | 3.1180 | 1264.46 | 0.67 | 5.75% |
| 26 | 16.9715 | 184.3400 | 2.9520 | 1470.84 | 0.65 | 5.55% |
| 27 | 17.5678 | 145.7637 | 3.1916 | 1344.71 | 0.72 | 6.25% |
| 28 | 17.7495 | 150.9267 | 3.1098 | 1456.91 | 0.70 | 5.72% |
| 29 | 14.5631 | 188.3314 | 2.9340 | 1325.59 | 0.63 | 5.53% |
| 30 | 17.7646 | 168.6067 | 3.3692 | 1326.16 | 0.74 | 6.27% |
| | | | | | | |
| Best | 12.8292 | 144.1521 | 2.7241 | 1254.00 | 0.59 | 4.55% |
| Worst | 17.7646 | 188.3314 | 3.6053 | 1470.84 | 0.77 | 6.56% |
| Average | 15.4584 | 160.0475 | 3.0946 | 1333.85 | 0.68 | 5.54% |
| Median | 15.3257 | 157.5513 | 3.0062 | 1306.64 | 0.68 | 5.54% |
| Std.Dev. | 1.3850 | 11.3541 | 0.2314 | 70.58 | 0.04 | 0.46% |

Table A.37 – MOPSO-e Statistics for Test Problem WEFR

| Run | GD | SP | iGD | PT | DR | DD |
|---|---|---|---|---|---|---|
| 1 | 10.9159 | 114.9281 | 3.2973 | 5830.28 | 0.44 | 5.66% |
| 2 | 9.0277 | 138.1650 | 3.0467 | 5776.44 | 0.49 | 4.51% |
| 3 | 8.6669 | 116.1389 | 3.8410 | 5755.96 | 0.45 | 4.32% |
| 4 | 10.9668 | 165.9537 | 3.0013 | 4709.37 | 0.61 | 5.41% |
| 5 | 8.7712 | 182.2637 | 4.0024 | 4731.08 | 0.35 | 5.21% |
| 6 | 9.1097 | 131.7171 | 5.5240 | 4828.59 | 0.61 | 4.22% |
| 7 | 9.9902 | 146.7281 | 3.1991 | 4793.88 | 0.56 | 4.57% |
| 8 | 7.1905 | 131.5049 | 3.7270 | 4732.09 | 0.43 | 3.87% |
| 9 | 12.3356 | 121.8152 | 3.6756 | 4825.91 | 0.57 | 4.37% |
| 10 | 7.4228 | 144.3999 | 2.9847 | 4639.60 | 0.56 | 4.42% |
| 11 | 11.7301 | 143.9152 | 2.9777 | 4722.65 | 0.57 | 4.31% |
| 12 | 9.9893 | 127.6164 | 3.4857 | 4761.33 | 0.51 | 3.98% |
| 13 | 8.1094 | 130.1783 | 3.5276 | 4661.47 | 0.47 | 4.19% |
| 14 | 7.9570 | 123.7542 | 3.9692 | 4755.66 | 0.49 | 4.38% |
| 15 | 7.4812 | 191.4700 | 3.8093 | 4673.71 | 0.47 | 4.77% |
| 16 | 8.0384 | 160.4258 | 3.5477 | 4666.83 | 0.51 | 4.17% |
| 17 | 10.9665 | 254.7030 | 3.4360 | 4594.01 | 0.49 | 5.21% |
| 18 | 11.5089 | 151.3082 | 3.2913 | 4701.62 | 0.50 | 5.34% |
| 19 | 11.3073 | 148.8493 | 3.3333 | 4641.09 | 0.53 | 4.69% |
| 20 | 13.4235 | 122.1707 | 3.0798 | 4789.14 | 0.55 | 4.62% |
| 21 | 19.0550 | 142.8925 | 3.1943 | 6051.32 | 0.55 | 4.47% |
| 22 | 8.6998 | 367.2743 | 3.9116 | 6151.96 | 0.47 | 4.86% |
| 23 | 7.8536 | 293.1238 | 3.1571 | 6894.47 | 0.49 | 3.76% |
| 24 | 9.7659 | 160.5531 | 3.3372 | 7010.59 | 0.53 | 4.25% |
| 25 | 7.4640 | 136.1343 | 3.8645 | 4716.88 | 0.50 | 4.17% |
| 26 | 7.5519 | 120.6106 | 5.0124 | 4656.41 | 0.48 | 4.50% |
| 27 | 7.9011 | 116.8589 | 3.5083 | 4617.12 | 0.45 | 4.68% |
| 28 | 7.1546 | 141.3810 | 3.2458 | 4655.82 | 0.46 | 3.86% |
| 29 | 8.0660 | 123.0953 | 3.0900 | 4687.21 | 0.53 | 4.94% |
| 30 | 10.9328 | 128.1746 | 3.1740 | 6052.79 | 0.53 | 4.32% |
| Best | 7.1546 | 114.9281 | 2.9777 | 4594.01 | 0.35 | 3.76% |
| Worst | 19.0550 | 367.2743 | 5.5240 | 7010.59 | 0.61 | 5.66% |
| Average | 9.6451 | 155.9368 | 3.5417 | 5102.84 | 0.50 | 4.53% |
| Median | 8.8995 | 139.7730 | 3.3866 | 4731.59 | 0.50 | 4.45% |
| Std.Dev. | 2.4807 | 55.9369 | 0.5680 | 711.80 | 0.06 | 0.48% |

Table A.38 – MOPSO-p Statistics for Test Problem WEFR

| Run | GD | SP | iGD | PT | DR | DD |
|---|---|---|---|---|---|---|
| 1 | 4.8958 | 159.2548 | 3.2641 | 1795.70 | 0.14 | 0.05% |
| 2 | 4.9434 | 125.4000 | 3.2184 | 1752.22 | 0.15 | 0.29% |
| 3 | 5.0003 | 143.6397 | 2.6039 | 1746.30 | 0.14 | 0.01% |
| 4 | 4.6807 | 129.6102 | 2.6462 | 1734.73 | 0.10 | 0.01% |
| 5 | 5.1570 | 243.7289 | 2.9229 | 1724.45 | 0.11 | 0.01% |
| 6 | 4.3311 | 146.3344 | 2.6851 | 1745.84 | 0.15 | 0.02% |
| 7 | 5.1548 | 142.9567 | 2.8080 | 1713.00 | 0.15 | 0.06% |
| 8 | 6.4960 | 160.4541 | 2.8355 | 1743.47 | 0.07 | 0.55% |
| 9 | 5.4412 | 162.5922 | 3.1563 | 1726.32 | 0.10 | 0.01% |
| 10 | 5.1238 | 127.1648 | 2.6883 | 1716.30 | 0.09 | 0.02% |
| 11 | 4.8136 | 133.9956 | 3.0033 | 1718.62 | 0.11 | 0.02% |
| 12 | 5.0724 | 227.0348 | 3.4268 | 1729.33 | 0.14 | 0.01% |
| 13 | 5.4865 | 220.8824 | 3.0157 | 1717.27 | 0.10 | 0.01% |
| 14 | 4.8440 | 120.9207 | 3.0067 | 1758.47 | 0.13 | 0.01% |
| 15 | 5.7138 | 259.4017 | 2.7658 | 1708.44 | 0.09 | 0.01% |
| 16 | 5.1067 | 125.1984 | 3.2343 | 1719.41 | 0.11 | 0.01% |
| 17 | 4.5221 | 250.2679 | 2.6217 | 1744.83 | 0.10 | 0.02% |
| 18 | 5.0896 | 141.2710 | 2.6555 | 1733.33 | 0.10 | 0.05% |
| 19 | 4.9151 | 140.0856 | 2.4685 | 1696.30 | 0.13 | 0.01% |
| 20 | 5.3847 | 240.9483 | 2.8855 | 1679.77 | 0.12 | 0.44% |
| 21 | 5.5047 | 244.7037 | 3.2914 | 1706.07 | 0.08 | 0.01% |
| 22 | 5.3830 | 111.0783 | 3.1439 | 1751.80 | 0.11 | 0.01% |
| 23 | 4.3902 | 152.0809 | 2.7726 | 1740.98 | 0.14 | 0.01% |
| 24 | 5.4377 | 126.7246 | 2.7500 | 1767.09 | 0.11 | 0.02% |
| 25 | 4.4431 | 110.7391 | 2.7755 | 1539.47 | 0.12 | 0.01% |
| 26 | 5.1157 | 120.2070 | 2.9636 | 1533.25 | 0.11 | 0.01% |
| 27 | 5.1099 | 149.2802 | 2.7758 | 1524.82 | 0.15 | 0.30% |
| 28 | 5.3134 | 162.2611 | 2.7393 | 1757.59 | 0.11 | 0.02% |
| 29 | 5.0798 | 217.1973 | 2.4859 | 1750.29 | 0.08 | 0.01% |
| 30 | 5.0478 | 131.5533 | 2.7430 | 1754.56 | 0.10 | 0.02% |
| | | | | | | |
| Best | 4.3311 | 110.7391 | 2.4685 | 1524.82 | 0.07 | 0.01% |
| Worst | 6.4960 | 259.4017 | 3.4268 | 1795.70 | 0.15 | 0.55% |
| Average | 5.0999 | 164.2323 | 2.8784 | 1714.33 | 0.12 | 0.07% |
| Median | 5.0981 | 144.9870 | 2.7919 | 1731.33 | 0.11 | 0.01% |
| Std.Dev. | 0.4308 | 47.8205 | 0.2503 | 65.83 | 0.02 | 0.14% |

Table A.39 – NSGA-II Statistics for Test Problem WEFR

| Run | GD | SP | iGD | PT | DR | DD |
|---|---|---|---|---|---|---|
| 1 | 11.8281 | 177.6477 | 2.6464 | 829.92 | 0.59 | 3.64% |
| 2 | 11.0739 | 168.5177 | 2.4675 | 829.22 | 0.50 | 3.50% |
| 3 | 9.6229 | 189.4582 | 2.7004 | 828.61 | 0.51 | 3.39% |
| 4 | 15.2689 | 203.7901 | 2.7729 | 828.00 | 0.50 | 4.90% |
| 5 | 13.4903 | 170.5897 | 2.4268 | 830.56 | 0.59 | 3.95% |
| 6 | 8.5458 | 140.9276 | 2.6092 | 829.67 | 0.53 | 4.72% |
| 7 | 9.1013 | 208.5959 | 2.7475 | 830.25 | 0.60 | 4.35% |
| 8 | 8.0484 | 125.2981 | 2.8702 | 828.34 | 0.43 | 3.13% |
| 9 | 7.4609 | 154.9180 | 2.6347 | 830.25 | 0.53 | 3.42% |
| 10 | 10.2083 | 164.2975 | 2.5443 | 830.45 | 0.54 | 4.22% |
| 11 | 7.9120 | 160.4953 | 2.7874 | 828.96 | 0.52 | 3.48% |
| 12 | 10.7095 | 199.3052 | 2.4197 | 829.89 | 0.53 | 3.48% |
| 13 | 8.6394 | 164.9697 | 2.8772 | 844.95 | 0.58 | 4.66% |
| 14 | 10.4942 | 147.0145 | 2.6930 | 841.64 | 0.54 | 5.05% |
| 15 | 12.9546 | 172.8155 | 2.7723 | 844.11 | 0.47 | 3.41% |
| 16 | 9.8623 | 139.3640 | 2.6369 | 845.31 | 0.54 | 4.36% |
| 17 | 9.7653 | 175.6649 | 2.5358 | 846.23 | 0.49 | 3.24% |
| 18 | 10.9363 | 170.2610 | 2.6766 | 845.81 | 0.53 | 4.36% |
| 19 | 7.8864 | 144.7998 | 2.6517 | 842.05 | 0.51 | 4.17% |
| 20 | 8.7703 | 184.1380 | 2.5904 | 842.33 | 0.53 | 4.17% |
| 21 | 12.3472 | 152.9824 | 2.7040 | 842.47 | 0.61 | 5.23% |
| 22 | 12.9019 | 175.5716 | 2.5719 | 842.60 | 0.58 | 4.52% |
| 23 | 12.5804 | 187.7406 | 2.6385 | 842.27 | 0.56 | 4.30% |
| 24 | 9.9072 | 203.9857 | 2.6268 | 842.08 | 0.53 | 3.21% |
| 25 | 7.8095 | 148.6638 | 2.7543 | 843.12 | 0.43 | 4.22% |
| 26 | 12.3258 | 169.0580 | 2.5312 | 842.18 | 0.52 | 4.00% |
| 27 | 10.4888 | 155.9387 | 2.5541 | 842.10 | 0.54 | 3.47% |
| 28 | 11.2599 | 168.6479 | 2.5954 | 842.46 | 0.54 | 3.40% |
| 29 | 14.9286 | 166.5014 | 2.5852 | 842.25 | 0.51 | 4.41% |
| 30 | 12.8359 | 161.8417 | 2.4042 | 856.83 | 0.53 | 3.57% |
| | | | | | | |
| Best | 7.4609 | 125.2981 | 2.4042 | 828.00 | 0.43 | 3.13% |
| Worst | 15.2689 | 208.5959 | 2.8772 | 856.83 | 0.61 | 5.23% |
| Average | 10.6655 | 168.4600 | 2.6342 | 838.16 | 0.53 | 4.00% |
| Median | 10.4915 | 168.5828 | 2.6358 | 842.09 | 0.53 | 4.09% |
| Std.Dev. | 2.1343 | 20.2949 | 0.1223 | 7.70 | 0.04 | 0.59% |

Table A.40 – RSearch Statistics for Test Problem WEFR

| Run | GD | SP | iGD | PT | DR | DD |
|---|---|---|---|---|---|---|
| 1 | 25.0994 | 160.0015 | 3.7432 | 1322.76 | 0.45 | 4.45% |
| 2 | 22.5793 | 159.2871 | 3.4119 | 1322.25 | 0.39 | 3.64% |
| 3 | 23.0496 | 171.0798 | 3.2490 | 1309.32 | 0.39 | 3.93% |
| 4 | 22.0225 | 171.3890 | 3.3362 | 1307.97 | 0.33 | 3.96% |
| 5 | 20.8738 | 161.5996 | 3.1569 | 1320.23 | 0.36 | 4.11% |
| 6 | 23.3853 | 179.6754 | 3.0643 | 1317.49 | 0.40 | 3.16% |
| 7 | 23.5337 | 195.8079 | 3.7463 | 1333.16 | 0.45 | 3.91% |
| 8 | 20.6414 | 171.7771 | 3.0385 | 1312.30 | 0.44 | 4.05% |
| 9 | 24.1996 | 156.3766 | 3.2255 | 1324.16 | 0.47 | 3.65% |
| 10 | 23.6857 | 197.7940 | 3.5631 | 1140.62 | 0.45 | 4.24% |
| 11 | 21.6958 | 165.4477 | 3.1765 | 1119.29 | 0.39 | 4.48% |
| 12 | 21.9521 | 164.9286 | 3.0076 | 1120.03 | 0.39 | 3.76% |
| 13 | 22.5030 | 162.6537 | 3.4630 | 1135.41 | 0.43 | 3.98% |
| 14 | 23.3554 | 161.4683 | 3.4016 | 1129.31 | 0.40 | 3.38% |
| 15 | 21.2570 | 187.8488 | 3.6765 | 1130.74 | 0.45 | 3.78% |
| 16 | 23.5661 | 187.8579 | 3.6188 | 1150.93 | 0.43 | 3.84% |
| 17 | 22.9391 | 183.3874 | 3.2063 | 1140.46 | 0.45 | 3.23% |
| 18 | 25.3357 | 189.9203 | 3.0982 | 1140.89 | 0.44 | 3.73% |
| 19 | 22.8388 | 173.9622 | 3.0594 | 1149.78 | 0.33 | 3.62% |
| 20 | 23.2672 | 185.4906 | 3.3811 | 1135.16 | 0.41 | 3.76% |
| 21 | 21.5301 | 161.5560 | 2.9398 | 1138.94 | 0.51 | 4.22% |
| 22 | 23.9864 | 169.6908 | 3.4404 | 1144.48 | 0.49 | 3.94% |
| 23 | 25.0402 | 179.5597 | 3.0839 | 1138.72 | 0.44 | 3.80% |
| 24 | 22.4183 | 203.0061 | 3.4242 | 1135.06 | 0.35 | 4.32% |
| 25 | 24.0417 | 202.5347 | 3.3591 | 1160.92 | 0.40 | 3.66% |
| 26 | 23.4918 | 169.5947 | 3.3039 | 1152.08 | 0.41 | 3.56% |
| 27 | 20.4519 | 177.0261 | 3.1284 | 1154.19 | 0.41 | 3.76% |
| 28 | 22.8669 | 190.0713 | 3.3103 | 1144.10 | 0.49 | 4.11% |
| 29 | 21.2875 | 188.8354 | 3.0120 | 1145.05 | 0.39 | 4.15% |
| 30 | 23.1979 | 167.5611 | 3.4714 | 1138.00 | 0.45 | 3.52% |
| | | | | | | |
| Best | 20.4519 | 156.3766 | 2.9398 | 1119.29 | 0.33 | 3.16% |
| Worst | 25.3357 | 203.0061 | 3.7463 | 1333.16 | 0.51 | 4.48% |
| Average | 22.8698 | 176.5730 | 3.3032 | 1193.79 | 0.42 | 3.86% |
| Median | 22.9943 | 172.8697 | 3.3071 | 1144.77 | 0.42 | 3.82% |
| Std.Dev. | 1.2864 | 13.7457 | 0.2261 | 83.82 | 0.04 | 0.33% |

Table A.41 – MOPSO Statistics for Test Problem TD

| Run | GD | SP | iGD | DI | PT | DR | DD |
|---|---|---|---|---|---|---|---|
| 1 | 0.060514 | 0.867595 | 0.114846 | 0.727724 | 155.80 | 0.14 | 0.24% |
| 2 | 0.062342 | 1.016362 | 0.086200 | 0.633157 | 151.00 | 0.09 | 0.54% |
| 3 | 0.055175 | 0.876487 | 0.073055 | 0.602200 | 152.50 | 0.10 | 0.39% |
| 4 | 0.092236 | 1.012393 | 0.171819 | 0.714858 | 147.53 | 0.17 | 0.95% |
| 5 | 0.062554 | 1.174246 | 0.106414 | 0.664298 | 154.28 | 0.12 | 0.54% |
| 6 | 0.073239 | 1.048066 | 0.101151 | 0.596437 | 148.86 | 0.19 | 0.62% |
| 7 | 0.081680 | 0.878357 | 0.086774 | 0.660574 | 149.30 | 0.19 | 0.98% |
| 8 | 0.113451 | 1.175527 | 0.090634 | 0.700021 | 146.31 | 0.22 | 0.41% |
| 9 | 0.058321 | 0.996310 | 0.110524 | 0.674600 | 151.42 | 0.20 | 0.20% |
| 10 | 0.058265 | 0.796791 | 0.094050 | 0.665756 | 154.06 | 0.13 | 0.20% |
| 11 | 0.063292 | 0.866557 | 0.087436 | 0.652248 | 156.16 | 0.24 | 0.36% |
| 12 | 0.062601 | 1.013742 | 0.108690 | 0.621686 | 156.02 | 0.14 | 0.27% |
| 13 | 0.070821 | 0.843163 | 0.113287 | 0.706387 | 142.14 | 0.27 | 0.34% |
| 14 | 0.058824 | 1.011879 | 0.089374 | 0.716726 | 152.80 | 0.14 | 0.57% |
| 15 | 0.058654 | 0.821712 | 0.092999 | 0.640609 | 148.09 | 0.16 | 0.22% |
| 16 | 0.060293 | 0.899022 | 0.164078 | 0.716460 | 154.77 | 0.24 | 0.24% |
| 17 | 0.059939 | 0.970647 | 0.103233 | 0.669707 | 147.24 | 0.20 | 0.26% |
| 18 | 0.050847 | 0.804646 | 0.155875 | 0.779778 | 157.69 | 0.15 | 0.44% |
| 19 | 0.057892 | 0.884552 | 0.100490 | 0.663812 | 155.45 | 0.14 | 0.22% |
| 20 | 0.063560 | 0.714238 | 0.095450 | 0.675389 | 157.22 | 0.16 | 0.35% |
| 21 | 0.061630 | 0.874271 | 0.110255 | 0.713517 | 142.75 | 0.22 | 0.50% |
| 22 | 0.061721 | 1.143016 | 0.098339 | 0.677148 | 153.66 | 0.16 | 0.34% |
| 23 | 0.064108 | 0.802352 | 0.083712 | 0.635555 | 159.45 | 0.22 | 0.46% |
| 24 | 0.060204 | 0.754176 | 0.147259 | 0.681516 | 150.17 | 0.20 | 0.20% |
| 25 | 0.098966 | 0.980907 | 0.100966 | 0.667620 | 154.70 | 0.24 | 0.85% |
| 26 | 0.066713 | 0.682635 | 0.131979 | 0.646037 | 152.02 | 0.13 | 0.16% |
| 27 | 0.153746 | 0.797082 | 0.102964 | 0.718192 | 155.16 | 0.22 | 0.56% |
| 28 | 0.063294 | 0.993033 | 0.120335 | 0.743025 | 149.13 | 0.17 | 0.26% |
| 29 | 0.057311 | 0.927410 | 0.089398 | 0.675520 | 149.81 | 0.20 | 0.21% |
| 30 | 0.056927 | 1.000196 | 0.097673 | 0.675811 | 149.33 | 0.25 | 0.35% |
| Best | 0.050847 | 0.682635 | 0.073055 | 0.596437 | 142.14 | 0.09 | 0.16% |
| Worst | 0.153746 | 1.175527 | 0.171819 | 0.779778 | 159.45 | 0.27 | 0.98% |
| Average | 0.068971 | 0.920912 | 0.107642 | 0.677212 | 151.83 | 0.18 | 0.41% |
| Median | 0.062031 | 0.891787 | 0.101058 | 0.674994 | 152.26 | 0.18 | 0.35% |
| Std.Dev. | 0.021017 | 0.126928 | 0.024209 | 0.041136 | 4.26 | 0.05 | 0.22% |

Table A.42 – MOPSO-e Statistics for Test Problem TD

| Run | GD | SP | iGD | DI | PT | DR | DD |
|---|---|---|---|---|---|---|---|
| 1 | 0.056832 | 0.950865 | 0.105099 | 0.660428 | 324.95 | 0.16 | 0.25% |
| 2 | 0.059508 | 0.820159 | 0.088016 | 0.629806 | 324.84 | 0.21 | 0.30% |
| 3 | 0.059229 | 1.025421 | 0.117138 | 0.746902 | 329.92 | 0.13 | 0.27% |
| 4 | 0.061786 | 0.837536 | 0.097096 | 0.678673 | 323.06 | 0.11 | 0.14% |
| 5 | 0.058566 | 0.844569 | 0.092114 | 0.622085 | 321.70 | 0.13 | 0.24% |
| 6 | 0.062672 | 0.733701 | 0.096711 | 0.728671 | 319.42 | 0.15 | 0.22% |
| 7 | 0.056959 | 1.132361 | 0.087844 | 0.646181 | 348.01 | 0.17 | 0.36% |
| 8 | 0.062438 | 0.752448 | 0.126587 | 0.635714 | 321.94 | 0.09 | 0.25% |
| 9 | 0.057244 | 0.832218 | 0.096185 | 0.628643 | 312.20 | 0.22 | 0.17% |
| 10 | 0.063833 | 0.884898 | 0.093143 | 0.662908 | 319.80 | 0.17 | 0.17% |
| 11 | 0.060047 | 0.798003 | 0.107454 | 0.682224 | 323.41 | 0.07 | 0.18% |
| 12 | 0.056610 | 0.844990 | 0.078064 | 0.575059 | 342.58 | 0.12 | 0.28% |
| 13 | 0.062676 | 0.916520 | 0.081754 | 0.661107 | 324.08 | 0.23 | 0.32% |
| 14 | 0.055459 | 1.008320 | 0.085289 | 0.710023 | 327.97 | 0.17 | 0.15% |
| 15 | 0.060439 | 0.716195 | 0.082209 | 0.602504 | 321.02 | 0.10 | 0.28% |
| 16 | 0.060125 | 0.667090 | 0.120875 | 0.654509 | 334.23 | 0.16 | 0.17% |
| 17 | 0.063196 | 0.631960 | 0.139123 | 0.686470 | 316.98 | 0.14 | 0.31% |
| 18 | 0.056701 | 0.987792 | 0.107456 | 0.739167 | 322.71 | 0.14 | 0.20% |
| 19 | 0.056715 | 0.907400 | 0.113395 | 0.710259 | 323.64 | 0.07 | 0.55% |
| 20 | 0.063584 | 0.718175 | 0.133039 | 0.713843 | 314.73 | 0.15 | 0.25% |
| 21 | 0.061195 | 0.748791 | 0.093781 | 0.627811 | 316.36 | 0.07 | 0.28% |
| 22 | 0.058607 | 0.859217 | 0.099632 | 0.715026 | 319.56 | 0.11 | 0.64% |
| 23 | 0.067583 | 0.840591 | 0.101470 | 0.647804 | 321.76 | 0.07 | 0.93% |
| 24 | 0.057218 | 0.833827 | 0.087561 | 0.646442 | 325.48 | 0.16 | 0.34% |
| 25 | 0.060479 | 0.918795 | 0.087229 | 0.690585 | 326.90 | 0.19 | 0.28% |
| 26 | 0.060101 | 0.880326 | 0.102323 | 0.715423 | 322.89 | 0.14 | 0.25% |
| 27 | 0.057145 | 1.132415 | 0.112708 | 0.726103 | 320.46 | 0.17 | 0.22% |
| 28 | 0.052310 | 0.749930 | 0.086930 | 0.674898 | 318.68 | 0.12 | 0.17% |
| 29 | 0.057756 | 0.839307 | 0.075617 | 0.648559 | 314.09 | 0.17 | 0.22% |
| 30 | 0.060771 | 0.906586 | 0.098586 | 0.665719 | 319.53 | 0.18 | 0.22% |
| | | | | | | | |
| Best | 0.052310 | 0.631960 | 0.075617 | 0.575059 | 312.20 | 0.07 | 0.14% |
| Worst | 0.067583 | 1.132415 | 0.139123 | 0.746902 | 348.01 | 0.23 | 0.93% |
| Average | 0.059593 | 0.857347 | 0.099814 | 0.671118 | 323.43 | 0.14 | 0.29% |
| Median | 0.059778 | 0.842580 | 0.096903 | 0.664314 | 322.32 | 0.15 | 0.25% |
| Std.Dev. | 0.003117 | 0.121045 | 0.016026 | 0.042398 | 7.56 | 0.04 | 0.16% |

Table A.43 – NSGA-II Statistics for Test Problem TD

| Run | GD | SP | iGD | DI | PT | DR | DD |
|---|---|---|---|---|---|---|---|
| 1 | 0.054356 | 0.781523 | 0.072570 | 0.540505 | 428.05 | 0.04 | 0.14% |
| 2 | 0.146390 | 1.434470 | 0.071110 | 0.650682 | 418.27 | 0.12 | 0.71% |
| 3 | 0.049158 | 0.757116 | 0.060101 | 0.517944 | 421.66 | 0.14 | 0.22% |
| 4 | 0.080633 | 0.722673 | 0.076068 | 0.564929 | 421.69 | 0.16 | 0.42% |
| 5 | 0.232775 | 0.726362 | 0.068211 | 0.612028 | 417.50 | 0.16 | 3.06% |
| 6 | 0.056565 | 0.758409 | 0.067195 | 0.525447 | 422.33 | 0.16 | 0.36% |
| 7 | 0.051376 | 0.779148 | 0.063719 | 0.539170 | 432.30 | 0.18 | 0.18% |
| 8 | 0.047820 | 0.754406 | 0.064577 | 0.537099 | 418.71 | 0.11 | 0.32% |
| 9 | 0.054688 | 0.820805 | 0.065766 | 0.521988 | 420.26 | 0.14 | 0.24% |
| 10 | 0.187977 | 1.768210 | 0.061916 | 0.553576 | 419.48 | 0.10 | 2.64% |
| 11 | 0.079721 | 0.930995 | 0.060653 | 0.542962 | 420.93 | 0.15 | 0.35% |
| 12 | 0.246202 | 0.857700 | 0.063698 | 0.675245 | 421.11 | 0.12 | 1.21% |
| 13 | 0.055296 | 0.744688 | 0.073831 | 0.512430 | 420.93 | 0.09 | 0.26% |
| 14 | 0.052201 | 0.700397 | 0.065410 | 0.514403 | 418.19 | 0.09 | 0.17% |
| 15 | 0.058368 | 0.746683 | 0.068134 | 0.559833 | 418.68 | 0.10 | 0.55% |
| 16 | 0.053608 | 0.716297 | 0.066914 | 0.483941 | 420.74 | 0.13 | 0.33% |
| 17 | 0.048499 | 0.727097 | 0.063498 | 0.529840 | 419.24 | 0.16 | 0.36% |
| 18 | 0.051812 | 0.776650 | 0.065319 | 0.513109 | 418.86 | 0.12 | 0.38% |
| 19 | 0.052261 | 0.642666 | 0.068039 | 0.495183 | 418.46 | 0.13 | 0.24% |
| 20 | 0.053610 | 0.631922 | 0.082733 | 0.546367 | 419.27 | 0.12 | 0.16% |
| 21 | 0.050129 | 0.751951 | 0.066482 | 0.561111 | 420.40 | 0.10 | 0.26% |
| 22 | 0.048053 | 0.764477 | 0.085718 | 0.538299 | 419.02 | 0.12 | 0.12% |
| 23 | 0.048878 | 0.825590 | 0.058478 | 0.512114 | 420.79 | 0.11 | 0.18% |
| 24 | 0.095376 | 0.745787 | 0.069388 | 0.503119 | 418.99 | 0.19 | 1.15% |
| 25 | 0.061160 | 0.628462 | 0.062890 | 0.460345 | 420.25 | 0.15 | 0.45% |
| 26 | 0.051528 | 0.751649 | 0.062825 | 0.501657 | 418.36 | 0.18 | 0.27% |
| 27 | 0.054264 | 0.717847 | 0.067728 | 0.497580 | 419.58 | 0.08 | 0.19% |
| 28 | 0.161004 | 0.970831 | 0.084460 | 0.680566 | 417.75 | 0.16 | 0.64% |
| 29 | 0.054690 | 0.721382 | 0.059564 | 0.482949 | 420.46 | 0.12 | 0.25% |
| 30 | 0.247424 | 2.474544 | 0.070096 | 0.669592 | 419.11 | 0.13 | 0.94% |
| | | | | | | | |
| Best | 0.047820 | 0.628462 | 0.058478 | 0.460345 | 417.50 | 0.04 | 0.12% |
| Worst | 0.247424 | 2.474544 | 0.085718 | 0.680566 | 432.30 | 0.19 | 3.06% |
| Average | 0.086194 | 0.871025 | 0.067903 | 0.544800 | 420.38 | 0.13 | 0.56% |
| Median | 0.054522 | 0.753179 | 0.066698 | 0.533470 | 419.53 | 0.12 | 0.33% |
| Std.Dev. | 0.063385 | 0.380301 | 0.006942 | 0.057751 | 2.99 | 0.03 | 0.68% |

Table A.44 – RSearch Statistics for Test Problem TD

| Run | GD | SP | iGD | DI | PT | DR | DD |
|---|---|---|---|---|---|---|---|
| 1 | 0.083268 | 1.890135 | 0.202563 | 0.696231 | 200.93 | 0.91 | 0.87% |
| 2 | 0.091882 | 0.637679 | 0.087113 | 0.539098 | 196.42 | 0.78 | 0.99% |
| 3 | 0.146039 | 0.769275 | 0.152654 | 0.573525 | 208.43 | 0.90 | 1.11% |
| 4 | 0.173530 | 0.918885 | 0.133003 | 0.629557 | 195.26 | 0.87 | 1.21% |
| 5 | 0.190238 | 0.818489 | 0.154227 | 0.632676 | 194.72 | 0.85 | 1.01% |
| 6 | 0.250359 | 0.853548 | 0.110851 | 0.667673 | 194.47 | 0.82 | 0.97% |
| 7 | 0.134149 | 1.215765 | 0.128195 | 0.641515 | 194.53 | 0.84 | 0.95% |
| 8 | 0.282471 | 1.432366 | 0.147676 | 0.680305 | 196.36 | 0.88 | 1.12% |
| 9 | 0.152105 | 1.016266 | 0.198408 | 0.647881 | 194.76 | 0.86 | 0.87% |
| 10 | 0.198065 | 1.811668 | 0.116284 | 0.648419 | 194.34 | 0.88 | 0.87% |
| 11 | 0.175915 | 1.134839 | 0.112543 | 0.648182 | 197.76 | 0.82 | 1.01% |
| 12 | 0.254391 | 0.818289 | 0.087550 | 0.598401 | 194.93 | 0.84 | 1.02% |
| 13 | 0.191018 | 1.382499 | 0.150364 | 0.712020 | 198.03 | 0.88 | 1.04% |
| 14 | 0.215020 | 1.376244 | 0.160133 | 0.723917 | 201.42 | 0.90 | 1.15% |
| 15 | 0.086597 | 0.993289 | 0.103182 | 0.549285 | 193.15 | 0.80 | 0.92% |
| 16 | 0.228642 | 0.987009 | 0.087783 | 0.598802 | 192.65 | 0.86 | 1.04% |
| 17 | 0.170075 | 1.228577 | 0.105100 | 0.617912 | 192.98 | 0.82 | 0.94% |
| 18 | 0.166072 | 0.834645 | 0.098314 | 0.601862 | 196.45 | 0.90 | 0.98% |
| 19 | 0.078440 | 1.220367 | 0.125275 | 0.620693 | 192.67 | 0.81 | 0.82% |
| 20 | 0.143247 | 1.219112 | 0.164674 | 0.635030 | 194.79 | 0.84 | 0.97% |
| 21 | 0.297144 | 0.734024 | 0.220527 | 0.684591 | 195.25 | 0.90 | 1.24% |
| 22 | 0.176747 | 0.984457 | 0.120706 | 0.613634 | 193.36 | 0.86 | 0.97% |
| 23 | 0.227383 | 1.126803 | 0.124666 | 0.682413 | 194.43 | 0.82 | 1.06% |
| 24 | 0.147900 | 0.979644 | 0.094152 | 0.616887 | 197.03 | 0.85 | 1.00% |
| 25 | 0.229005 | 1.919949 | 0.127125 | 0.689975 | 195.07 | 0.89 | 0.96% |
| 26 | 0.107867 | 0.769013 | 0.130396 | 0.610225 | 202.54 | 0.85 | 0.99% |
| 27 | 0.226197 | 0.820063 | 0.128041 | 0.626778 | 195.20 | 0.86 | 1.21% |
| 28 | 0.322975 | 1.180788 | 0.088016 | 0.651466 | 195.04 | 0.90 | 1.22% |
| 29 | 0.107264 | 0.765553 | 0.165400 | 0.632342 | 196.14 | 0.86 | 0.88% |
| 30 | 0.074890 | 1.069071 | 0.105733 | 0.606380 | 195.03 | 0.88 | 0.89% |
| | | | | | | | |
| Best | 0.074890 | 0.637679 | 0.087113 | 0.539098 | 192.65 | 0.78 | 0.82% |
| Worst | 0.322975 | 1.919949 | 0.220527 | 0.723917 | 208.43 | 0.91 | 1.24% |
| Average | 0.177630 | 1.096944 | 0.131022 | 0.635923 | 196.14 | 0.86 | 1.01% |
| Median | 0.174722 | 1.004778 | 0.126200 | 0.632509 | 195.06 | 0.86 | 0.99% |
| Std.Dev. | 0.067052 | 0.336094 | 0.035098 | 0.043999 | 3.35 | 0.03 | 0.11% |

Table A.45 – MOPSO Statistics for Test Problem TDT

| Run | GD | SP | iGD | PT | DR | DD |
|---|---|---|---|---|---|---|
| 1 | 0.433798 | 2.773365 | 0.042930 | 901.31 | 0.91 | 5.45% |
| 2 | 0.384636 | 2.355714 | 0.039054 | 912.20 | 0.91 | 5.25% |
| 3 | 0.431199 | 2.472662 | 0.039373 | 882.82 | 0.90 | 6.30% |
| 4 | 0.450736 | 2.175379 | 0.039980 | 887.30 | 0.92 | 4.28% |
| 5 | 0.461465 | 2.398247 | 0.050938 | 884.80 | 0.93 | 4.99% |
| 6 | 0.442248 | 2.608800 | 0.039509 | 890.12 | 0.91 | 5.12% |
| 7 | 0.459009 | 2.337184 | 0.048792 | 895.73 | 0.95 | 6.24% |
| 8 | 0.391887 | 2.258209 | 0.043485 | 876.76 | 0.91 | 3.96% |
| 9 | 0.442933 | 2.678308 | 0.046746 | 877.63 | 0.95 | 4.10% |
| 10 | 0.408106 | 2.505391 | 0.042079 | 878.01 | 0.94 | 5.03% |
| 11 | 0.392643 | 2.386618 | 0.042660 | 892.58 | 0.93 | 3.96% |
| 12 | 0.473238 | 2.430618 | 0.044403 | 884.00 | 0.93 | 5.41% |
| 13 | 0.451068 | 2.081319 | 0.039521 | 875.58 | 0.93 | 3.84% |
| 14 | 0.415740 | 2.313618 | 0.038101 | 894.88 | 0.86 | 5.15% |
| 15 | 0.394051 | 2.336478 | 0.042502 | 890.99 | 0.91 | 3.52% |
| 16 | 0.395234 | 2.602495 | 0.039949 | 891.83 | 0.93 | 4.51% |
| 17 | 0.430600 | 2.418658 | 0.044123 | 891.25 | 0.93 | 5.20% |
| 18 | 0.409628 | 2.644974 | 0.042951 | 891.67 | 0.91 | 4.67% |
| 19 | 0.431878 | 2.272540 | 0.043542 | 897.34 | 0.94 | 4.56% |
| 20 | 0.451906 | 2.199790 | 0.046072 | 882.45 | 0.95 | 5.33% |
| 21 | 0.420267 | 2.499283 | 0.040221 | 896.24 | 0.91 | 4.45% |
| 22 | 0.437653 | 2.498566 | 0.046601 | 876.45 | 0.91 | 6.27% |
| 23 | 0.450572 | 2.674243 | 0.042766 | 874.41 | 0.93 | 5.81% |
| 24 | 0.414022 | 2.465968 | 0.042902 | 878.59 | 0.93 | 5.66% |
| 25 | 0.378901 | 2.310994 | 0.040675 | 872.19 | 0.93 | 5.08% |
| 26 | 0.421476 | 2.417644 | 0.038555 | 910.64 | 0.96 | 3.62% |
| 27 | 0.425068 | 2.478898 | 0.041583 | 893.41 | 0.93 | 4.97% |
| 28 | 0.392636 | 2.624482 | 0.041950 | 885.15 | 0.93 | 4.14% |
| 29 | 0.419847 | 2.503934 | 0.041948 | 897.02 | 0.93 | 5.01% |
| 30 | 0.407551 | 2.191467 | 0.042464 | 898.58 | 0.92 | 4.40% |
| | | | | | | |
| Best | 0.378901 | 2.081319 | 0.038101 | 872.19 | 0.86 | 3.52% |
| Worst | 0.473238 | 2.773365 | 0.050938 | 912.20 | 0.96 | 6.30% |
| Average | 0.424000 | 2.430528 | 0.042546 | 888.73 | 0.92 | 4.88% |
| Median | 0.423272 | 2.424638 | 0.042483 | 890.55 | 0.93 | 5.00% |
| Std.Dev. | 0.025161 | 0.167691 | 0.003021 | 10.26 | 0.02 | 0.77% |

Table A.46 – MOPSO-e Statistics for Test Problem TDT

| Run | GD | SP | iGD | PT | DR | DD |
|---|---|---|---|---|---|---|
| 1 | 0.46121 | 2.68133 | 0.04333 | 1649.51 | 0.77 | 7.47% |
| 2 | 0.41578 | 2.80684 | 0.04052 | 1659.68 | 0.77 | 6.23% |
| 3 | 0.44326 | 2.41046 | 0.04572 | 1650.98 | 0.75 | 5.62% |
| 4 | 0.42655 | 2.43439 | 0.04347 | 1665.06 | 0.78 | 4.74% |
| 5 | 0.46880 | 3.05575 | 0.04119 | 1650.51 | 0.79 | 5.88% |
| 6 | 0.41483 | 2.53190 | 0.04551 | 1666.01 | 0.74 | 4.62% |
| 7 | 0.40011 | 2.31032 | 0.04216 | 1617.78 | 0.78 | 5.45% |
| 8 | 0.48274 | 2.46727 | 0.04497 | 1632.73 | 0.73 | 6.43% |
| 9 | 0.40470 | 2.41904 | 0.04534 | 1649.12 | 0.76 | 4.70% |
| 10 | 0.47796 | 2.53220 | 0.04393 | 1621.49 | 0.75 | 6.61% |
| 11 | 0.43492 | 2.43622 | 0.04494 | 1650.22 | 0.80 | 7.13% |
| 12 | 0.38038 | 2.20491 | 0.04254 | 1670.15 | 0.78 | 4.99% |
| 13 | 0.39477 | 2.12733 | 0.04476 | 1644.38 | 0.79 | 5.47% |
| 14 | 0.48511 | 2.76270 | 0.05058 | 1682.99 | 0.81 | 6.53% |
| 15 | 0.40133 | 2.12844 | 0.04061 | 1660.35 | 0.74 | 4.11% |
| 16 | 0.40846 | 2.26850 | 0.04318 | 1658.46 | 0.73 | 4.23% |
| 17 | 0.39105 | 2.58648 | 0.03969 | 1662.12 | 0.74 | 5.62% |
| 18 | 0.42520 | 2.48931 | 0.04126 | 1644.72 | 0.73 | 5.19% |
| 19 | 0.47876 | 2.58487 | 0.04287 | 1605.88 | 0.77 | 5.34% |
| 20 | 0.43631 | 2.47331 | 0.04971 | 1634.21 | 0.83 | 7.29% |
| 21 | 0.38079 | 2.37358 | 0.04109 | 1628.01 | 0.74 | 4.99% |
| 22 | 0.41383 | 2.40608 | 0.03809 | 1615.64 | 0.69 | 6.57% |
| 23 | 0.43454 | 2.56380 | 0.04266 | 1658.31 | 0.76 | 4.92% |
| 24 | 0.39044 | 2.50270 | 0.04695 | 1635.48 | 0.75 | 5.82% |
| 25 | 0.43381 | 2.40941 | 0.04271 | 1610.66 | 0.75 | 5.80% |
| 26 | 0.45639 | 2.36473 | 0.04429 | 1657.28 | 0.69 | 6.43% |
| 27 | 0.43062 | 2.50254 | 0.04599 | 1650.13 | 0.77 | 5.89% |
| 28 | 0.45287 | 2.21368 | 0.04011 | 1611.74 | 0.71 | 5.74% |
| 29 | 0.46410 | 2.40422 | 0.04118 | 1669.87 | 0.71 | 5.76% |
| 30 | 0.46092 | 2.73719 | 0.04145 | 1678.32 | 0.78 | 6.83% |
| | | | | | | |
| Best | 0.38038 | 2.12733 | 0.03809 | 1605.88 | 0.69 | 4.11% |
| Worst | 0.48511 | 3.05575 | 0.05058 | 1682.99 | 0.83 | 7.47% |
| Average | 0.43168 | 2.47298 | 0.04336 | 1646.39 | 0.76 | 5.75% |
| Median | 0.43221 | 2.45175 | 0.04303 | 1650.17 | 0.76 | 5.75% |
| Std.Dev. | 0.03176 | 0.20180 | 0.00280 | 20.86 | 0.03 | 0.88% |

Table A.47 – NSGA-II Statistics for Test Problem TDT

| Run | GD | SP | iGD | PT | DR | DD |
|---|---|---|---|---|---|---|
| 1 | 0.17234 | 2.25219 | 0.05452 | 882.72 | 0.88 | 8.77% |
| 2 | 0.23046 | 2.79510 | 0.04491 | 882.79 | 0.93 | 9.41% |
| 3 | 0.25383 | 2.89305 | 0.05729 | 883.48 | 0.95 | 9.52% |
| 4 | 0.26102 | 2.86412 | 0.04757 | 882.55 | 0.90 | 9.52% |
| 5 | 0.23528 | 2.42298 | 0.05287 | 880.30 | 0.93 | 8.38% |
| 6 | 0.22271 | 2.63100 | 0.04537 | 944.26 | 0.94 | 8.75% |
| 7 | 0.24364 | 3.08657 | 0.05175 | 935.95 | 0.88 | 9.94% |
| 8 | 0.18584 | 2.39422 | 0.04429 | 938.55 | 0.90 | 6.98% |
| 9 | 0.23670 | 2.71480 | 0.05176 | 936.61 | 0.96 | 9.91% |
| 10 | 0.18233 | 2.36743 | 0.04816 | 940.15 | 0.93 | 8.52% |
| 11 | 0.14666 | 1.68671 | 0.04689 | 879.66 | 0.80 | 9.66% |
| 12 | 0.17218 | 2.03460 | 0.04316 | 870.42 | 0.77 | 8.33% |
| 13 | 0.22289 | 2.35028 | 0.04850 | 868.73 | 0.80 | 10.76% |
| 14 | 0.20978 | 2.85956 | 0.05125 | 877.00 | 0.80 | 8.81% |
| 15 | 0.21540 | 2.36032 | 0.04657 | 870.31 | 0.81 | 10.38% |
| 16 | 0.20154 | 2.23060 | 0.04938 | 968.30 | 0.93 | 9.33% |
| 17 | 0.23323 | 2.86031 | 0.04210 | 905.10 | 0.91 | 8.38% |
| 18 | 0.21210 | 1.94135 | 0.04651 | 907.55 | 0.95 | 7.71% |
| 19 | 0.23485 | 2.42323 | 0.04900 | 907.87 | 0.93 | 9.81% |
| 20 | 0.26007 | 2.18106 | 0.05544 | 907.25 | 0.95 | 11.23% |
| 21 | 0.26599 | 2.59818 | 0.06079 | 959.84 | 0.95 | 9.28% |
| 22 | 0.20235 | 2.63155 | 0.04985 | 930.97 | 0.87 | 7.85% |
| 23 | 0.18500 | 2.76959 | 0.04529 | 932.46 | 0.89 | 8.47% |
| 24 | 0.19749 | 2.52061 | 0.04720 | 933.11 | 0.89 | 9.67% |
| 25 | 0.20506 | 2.57760 | 0.04500 | 931.68 | 0.95 | 8.28% |
| 26 | 0.22676 | 2.34653 | 0.05279 | 956.70 | 0.92 | 9.88% |
| 27 | 0.19589 | 2.43932 | 0.04926 | 920.82 | 0.93 | 8.95% |
| 28 | 0.24456 | 3.30016 | 0.05050 | 925.40 | 0.94 | 8.95% |
| 29 | 0.21878 | 2.22330 | 0.05479 | 920.91 | 0.95 | 9.90% |
| 30 | 0.22166 | 2.19281 | 0.05081 | 921.18 | 0.93 | 9.57% |
| | | | | | | |
| Best | 0.14666 | 1.68671 | 0.04210 | 868.73 | 0.77 | 6.98% |
| Worst | 0.26599 | 3.30016 | 0.06079 | 968.30 | 0.96 | 11.23% |
| Average | 0.21655 | 2.49830 | 0.04945 | 913.42 | 0.90 | 9.16% |
| Median | 0.22022 | 2.43127 | 0.04913 | 920.86 | 0.93 | 9.31% |
| Std.Dev. | 0.02897 | 0.34708 | 0.00435 | 29.51 | 0.05 | 0.92% |

Table A.48 – RSearch Statistics for Test Problem TDT

| Run | GD | SP | iGD | PT | DR | DD |
|---|---|---|---|---|---|---|
| 1 | 0.42758 | 2.17832 | 0.22409 | 312.59 | 1.00 | 10.60% |
| 2 | 0.44914 | 2.24548 | 0.17654 | 334.25 | 1.00 | 9.43% |
| 3 | 0.50110 | 2.40208 | 0.22722 | 315.21 | 0.99 | 10.27% |
| 4 | 0.49173 | 3.80138 | 0.11932 | 315.17 | 0.99 | 11.79% |
| 5 | 0.54171 | 2.82608 | 0.13536 | 312.79 | 0.99 | 11.36% |
| 6 | 0.57026 | 2.01595 | 0.23532 | 328.90 | 1.00 | 9.50% |
| 7 | 0.47524 | 2.58518 | 0.17407 | 329.28 | 0.99 | 9.49% |
| 8 | 0.53515 | 2.69137 | 0.19645 | 319.00 | 0.99 | 10.49% |
| 9 | 0.42454 | 3.60500 | 0.12773 | 318.44 | 1.00 | 11.07% |
| 10 | 0.45304 | 9.29300 | 0.20215 | 318.02 | 0.99 | 10.38% |
| 11 | 0.44257 | 2.28796 | 0.16164 | 314.69 | 1.00 | 10.72% |
| 12 | 0.48112 | 2.20072 | 0.16351 | 333.26 | 0.99 | 9.44% |
| 13 | 0.48418 | 2.73765 | 0.12632 | 319.34 | 1.00 | 10.84% |
| 14 | 0.50412 | 6.89565 | 0.13853 | 319.41 | 0.99 | 9.68% |
| 15 | 0.46806 | 1.92277 | 0.19686 | 313.27 | 1.00 | 10.77% |
| 16 | 0.48939 | 2.41012 | 0.25344 | 311.49 | 0.99 | 10.38% |
| 17 | 0.55029 | 2.29339 | 0.16220 | 311.84 | 1.00 | 9.53% |
| 18 | 0.47566 | 3.47047 | 0.15191 | 314.55 | 1.00 | 9.39% |
| 19 | 0.49872 | 3.25750 | 0.14901 | 308.93 | 0.98 | 10.20% |
| 20 | 0.47075 | 1.89901 | 0.19453 | 312.30 | 0.99 | 9.62% |
| 21 | 0.46432 | 8.10425 | 0.17121 | 309.44 | 0.99 | 9.67% |
| 22 | 0.41667 | 2.01718 | 0.21495 | 310.20 | 0.99 | 9.79% |
| 23 | 0.47210 | 3.10271 | 0.14035 | 328.07 | 0.99 | 9.72% |
| 24 | 0.40629 | 2.85148 | 0.20730 | 323.96 | 0.99 | 11.71% |
| 25 | 0.52898 | 8.62005 | 0.17195 | 303.70 | 0.99 | 9.93% |
| 26 | 0.50490 | 6.95080 | 0.13229 | 314.74 | 0.99 | 8.93% |
| 27 | 0.52437 | 2.79277 | 0.23162 | 310.94 | 0.99 | 9.43% |
| 28 | 0.48524 | 9.71542 | 0.21426 | 313.98 | 1.00 | 9.61% |
| 29 | 0.49292 | 2.29910 | 0.20972 | 328.20 | 0.99 | 10.52% |
| 30 | 0.48555 | 2.70297 | 0.22359 | 347.52 | 1.00 | 10.18% |
| | | | | | | |
| Best | 0.40629 | 1.89901 | 0.11932 | 303.70 | 0.98 | 8.93% |
| Worst | 0.57026 | 9.71542 | 0.25344 | 347.52 | 1.00 | 11.79% |
| Average | 0.48386 | 3.73919 | 0.18111 | 318.45 | 0.99 | 10.15% |
| Median | 0.48471 | 2.72031 | 0.17530 | 314.96 | 0.99 | 10.06% |
| Std.Dev. | 0.03950 | 2.39935 | 0.03813 | 9.44 | 0.01 | 0.73% |

Table A.49 – MOPSO Statistics for Test Problem TDT2

| Run | GD | SP | iGD | PT | DR | DD |
|---|---|---|---|---|---|---|
| 1 | 0.17584 | 1.85720 | 0.00371 | 4982.44 | 0.14 | 4.97% |
| 2 | 0.00033 | 0.28655 | 0.00325 | 7641.58 | 0.12 | 0.01% |
| 3 | 0.00448 | 0.26491 | 0.00321 | 6274.35 | 0.13 | 3.40% |
| 4 | 0.00179 | 0.27362 | 0.00367 | 6289.31 | 0.14 | 0.70% |
| 5 | 0.00094 | 0.32586 | 0.00318 | 11822.00 | 0.12 | 0.82% |
| 6 | 0.00138 | 0.21005 | 0.00467 | 7031.29 | 0.16 | 0.19% |
| 7 | 0.00603 | 0.35051 | 0.00331 | 7081.95 | 0.11 | 0.61% |
| 8 | 0.00033 | 0.26742 | 0.00298 | 7565.97 | 0.10 | 0.00% |
| 9 | 0.00121 | 0.27222 | 0.00341 | 7458.34 | 0.07 | 0.07% |
| 10 | 0.00254 | 0.26970 | 0.00270 | 7388.59 | 0.14 | 2.41% |
| 11 | 0.00065 | 0.32295 | 0.00343 | 6941.00 | 0.09 | 0.03% |
| 12 | 0.00060 | 0.26761 | 0.00356 | 7953.29 | 0.11 | 0.01% |
| 13 | 0.10152 | 1.23371 | 0.00417 | 5928.40 | 0.09 | 11.57% |
| 14 | 0.00087 | 0.31714 | 0.00318 | 7755.36 | 0.13 | 2.53% |
| 15 | 0.36112 | 1.11968 | 0.00388 | 8460.48 | 0.11 | 37.46% |
| 16 | 0.00022 | 0.21478 | 0.00476 | 5386.28 | 0.11 | 0.00% |
| 17 | 0.00078 | 0.24462 | 0.00389 | 5798.43 | 0.07 | 0.00% |
| 18 | 0.00064 | 0.28633 | 0.00388 | 7584.26 | 0.13 | 0.00% |
| 19 | 0.00059 | 0.23773 | 0.00403 | 5807.71 | 0.14 | 0.01% |
| 20 | 0.00195 | 0.26151 | 0.00341 | 5748.26 | 0.16 | 0.65% |
| 21 | 0.00049 | 0.34376 | 0.00319 | 7636.82 | 0.08 | 0.01% |
| 22 | 0.24705 | 0.31861 | 0.00405 | 7459.35 | 0.14 | 12.21% |
| 23 | 0.00127 | 0.29538 | 0.00337 | 8139.89 | 0.11 | 0.55% |
| 24 | 0.24749 | 0.23838 | 0.00364 | 5521.78 | 0.20 | 6.92% |
| 25 | 0.46727 | 6.57108 | 0.00489 | 8089.89 | 0.08 | 29.03% |
| 26 | 0.00053 | 0.33238 | 0.00383 | 5459.57 | 0.09 | 0.00% |
| 27 | 0.00015 | 0.30512 | 0.00298 | 5834.52 | 0.13 | 0.00% |
| 28 | 0.00039 | 0.29379 | 0.00326 | 5651.23 | 0.12 | 0.00% |
| 29 | 0.00024 | 0.31607 | 0.00370 | 5889.05 | 0.08 | 0.00% |
| 30 | 0.00278 | 0.25299 | 0.00464 | 5903.94 | 0.12 | 0.49% |
| | | | | | | |
| Best | 0.00015 | 0.21005 | 0.00270 | 4982.44 | 0.07 | 0.00% |
| Worst | 0.46727 | 6.57108 | 0.00489 | 11822.00 | 0.20 | 37.46% |
| Average | 0.05438 | 0.60505 | 0.00366 | 6882.84 | 0.12 | 3.82% |
| Median | 0.00107 | 0.29017 | 0.00360 | 6986.15 | 0.12 | 0.34% |
| Std.Dev. | 0.12086 | 1.18244 | 0.00055 | 1370.37 | 0.03 | 8.69% |

Table A.50 – NSGA-II Statistics for Test Problem TDT2

| Run | GD | SP | iGD | PT | DR | DD |
|---|---|---|---|---|---|---|
| 1 | 0.00007 | 0.31409 | 0.00443 | 9237.67 | 0.11 | 0.00% |
| 2 | 0.24841 | 0.40195 | 0.00474 | 10093.77 | 0.12 | 11.58% |
| 3 | 0.00009 | 0.20687 | 0.00826 | 11684.67 | 0.12 | 0.01% |
| 4 | 0.00014 | 0.32027 | 0.00417 | 10629.56 | 0.16 | 2.05% |
| 5 | 0.80931 | 0.29172 | 0.00828 | 9426.71 | 0.21 | 31.57% |
| 6 | 0.24680 | 0.39834 | 0.00513 | 10091.75 | 0.12 | 14.28% |
| 7 | 0.00006 | 0.34352 | 0.00563 | 9675.02 | 0.17 | 0.00% |
| 8 | 0.00009 | 0.27494 | 0.00677 | 10483.24 | 0.16 | 0.00% |
| 9 | 0.00008 | 0.36411 | 0.00598 | 10066.00 | 0.19 | 0.00% |
| 10 | 0.00103 | 0.35411 | 0.00491 | 10279.89 | 0.14 | 2.37% |
| 11 | 0.00013 | 0.30480 | 0.00568 | 10192.32 | 0.10 | 0.01% |
| 12 | 0.13224 | 0.27990 | 0.00382 | 10540.78 | 0.14 | 14.94% |
| 13 | 0.00003 | 0.33013 | 0.00570 | 10805.46 | 0.17 | 0.00% |
| 14 | 0.00011 | 0.17557 | 0.00811 | 11308.10 | 0.15 | 0.00% |
| 15 | 0.00013 | 0.29249 | 0.00394 | 11029.64 | 0.10 | 0.00% |
| 16 | 0.00011 | 0.24217 | 0.00654 | 11177.49 | 0.10 | 0.01% |
| 17 | 0.00017 | 0.27195 | 0.00525 | 10406.57 | 0.11 | 0.00% |
| 18 | 0.00009 | 0.27835 | 0.00395 | 11830.51 | 0.13 | 0.00% |
| 19 | 0.00153 | 0.34056 | 0.00506 | 9296.54 | 0.16 | 0.64% |
| 20 | 0.00013 | 0.28167 | 0.00440 | 11216.70 | 0.08 | 0.01% |
| 21 | 0.66083 | 0.25450 | 0.00932 | 10814.70 | 0.12 | 36.84% |
| 22 | 0.00015 | 0.34931 | 0.00589 | 10910.55 | 0.13 | 0.00% |
| 23 | 0.00216 | 0.38802 | 0.00378 | 10455.64 | 0.10 | 6.64% |
| 24 | 0.00022 | 0.20727 | 0.00792 | 11239.65 | 0.16 | 0.00% |
| 25 | 0.34966 | 0.26415 | 0.00813 | 10074.62 | 0.14 | 19.81% |
| 26 | 0.14339 | 0.20281 | 0.00810 | 10141.80 | 0.15 | 13.85% |
| 27 | 0.30333 | 0.31832 | 0.00571 | 10865.45 | 0.16 | 13.00% |
| 28 | 0.00034 | 0.32514 | 0.00451 | 10760.78 | 0.15 | 0.00% |
| 29 | 0.00042 | 0.39721 | 0.00522 | 9463.82 | 0.09 | 3.66% |
| 30 | 0.00003 | 0.26747 | 0.00610 | 10142.20 | 0.15 | 0.00% |
| | | | | | | |
| Best | 0.00003 | 0.17557 | 0.00378 | 9237.67 | 0.08 | 0.00% |
| Worst | 0.80931 | 0.40195 | 0.00932 | 11830.51 | 0.21 | 36.84% |
| Average | 0.09671 | 0.30139 | 0.00585 | 10478.05 | 0.14 | 5.71% |
| Median | 0.00016 | 0.29864 | 0.00565 | 10469.44 | 0.14 | 0.01% |
| Std.Dev. | 0.20223 | 0.06029 | 0.00159 | 676.01 | 0.03 | 9.75% |

Table A.51 – RSearch Statistics for Test Problem TDT2

| Run | GD | SP | iGD | PT | DR | DD |
|---|---|---|---|---|---|---|
| 1 | 0.20127 | 1.13173 | 0.00432 | 14780.75 | 0.07 | 22.86% |
| 2 | 0.30325 | 0.34436 | 0.02899 | 12985.38 | 0.05 | 41.53% |
| 3 | 0.58330 | 5.07316 | 0.01188 | 15449.24 | 0.11 | 63.39% |
| 4 | 0.09449 | 1.18410 | 0.00589 | 13815.17 | 0.06 | 22.70% |
| 5 | 0.33496 | 1.44279 | 0.00949 | 11952.29 | 0.08 | 49.20% |
| 6 | 0.56024 | 0.83886 | 0.00526 | 15581.39 | 0.14 | 72.36% |
| 7 | 0.52716 | 5.05949 | 0.01713 | 12767.43 | 0.08 | 76.61% |
| 8 | 0.55704 | 4.94761 | 0.01636 | 11659.52 | 0.06 | 71.46% |
| 9 | 0.92178 | 1.44522 | 0.01122 | 12523.70 | 0.11 | 106.07% |
| 10 | 0.00967 | 0.49192 | 0.00943 | 12447.38 | 0.03 | 16.28% |
| 11 | 0.82127 | 3.77808 | 0.00435 | 12338.96 | 0.09 | 106.54% |
| 12 | 0.26757 | 1.23552 | 0.01972 | 12336.85 | 0.07 | 39.34% |
| 13 | 0.24485 | 1.87132 | 0.02464 | 12942.37 | 0.06 | 69.14% |
| 14 | 0.26760 | 1.23610 | 0.01287 | 13273.62 | 0.04 | 60.58% |
| 15 | 0.53805 | 5.00977 | 0.02232 | 13677.67 | 0.07 | 60.83% |
| 16 | 0.65120 | 5.48073 | 0.00452 | 12163.14 | 0.07 | 94.44% |
| 17 | 0.30324 | 0.28079 | 0.03219 | 14866.03 | 0.05 | 41.70% |
| 18 | 0.57521 | 4.94840 | 0.02336 | 14434.78 | 0.08 | 79.58% |
| 19 | 0.84550 | 3.64232 | 0.01534 | 13469.82 | 0.09 | 109.91% |
| 20 | 0.94361 | 0.84036 | 0.02083 | 11750.70 | 0.13 | 89.86% |
| 21 | 0.62437 | 5.09477 | 0.00534 | 15675.10 | 0.10 | 75.20% |
| 22 | 0.55315 | 4.37345 | 0.01281 | 14724.48 | 0.13 | 68.72% |
| 23 | 0.52885 | 4.95290 | 0.01911 | 14250.34 | 0.05 | 71.98% |
| 24 | 0.94338 | 1.19630 | 0.01268 | 12455.70 | 0.10 | 130.14% |
| 25 | 0.55678 | 5.01959 | 0.02126 | 12410.03 | 0.10 | 66.66% |
| 26 | 0.26624 | 0.81717 | 0.00481 | 12223.24 | 0.04 | 51.08% |
| 27 | 0.33312 | 0.33615 | 0.01363 | 12476.68 | 0.09 | 49.96% |
| 28 | 0.53800 | 5.01680 | 0.02231 | 12423.58 | 0.06 | 70.89% |
| 29 | 0.30343 | 0.26996 | 0.03201 | 13648.92 | 0.08 | 32.09% |
| 30 | 0.65922 | 5.37457 | 0.01384 | 13992.34 | 0.07 | 100.93% |
| | | | | | | |
| Best | 0.00967 | 0.26996 | 0.00432 | 11659.52 | 0.03 | 16.28% |
| Worst | 0.94361 | 5.48073 | 0.03219 | 15675.10 | 0.14 | 130.14% |
| Average | 0.49526 | 2.75781 | 0.01526 | 13316.55 | 0.08 | 67.07% |
| Median | 0.53802 | 1.65827 | 0.01374 | 12963.88 | 0.07 | 68.93% |
| Std.Dev. | 0.24774 | 2.04451 | 0.00827 | 1194.28 | 0.03 | 27.83% |

## APPENDIX B: GRAPHS OF THE SENSITIVITY ANALYSIS

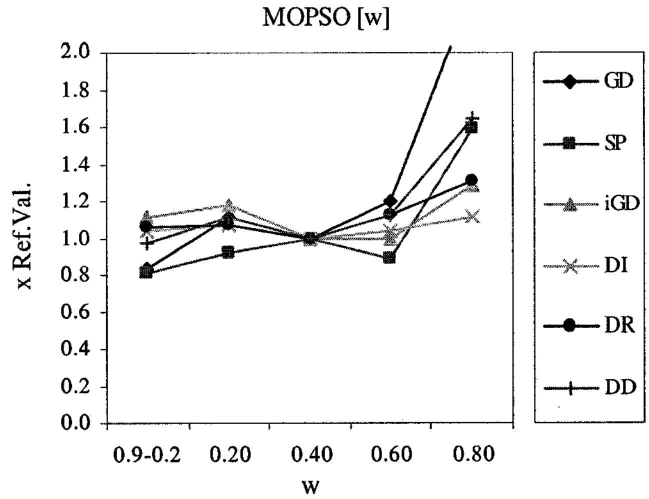



Figure B.1 – Sensitivity Analysis for Parameter *w* for Test Problem 1

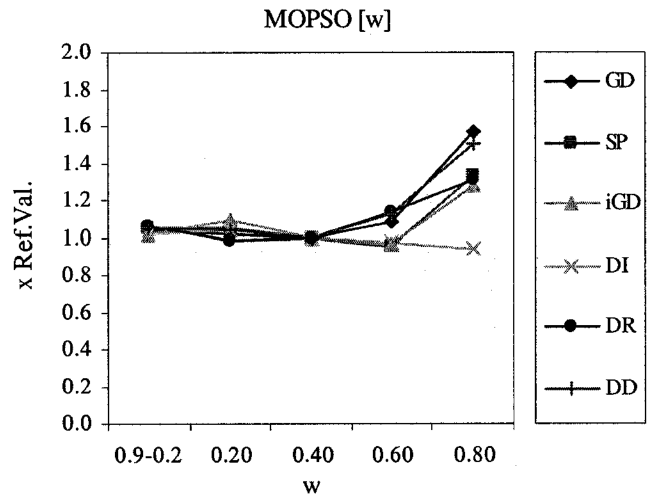



Figure B.2 – Sensitivity Analysis for Parameter *w* for Test Problem 5

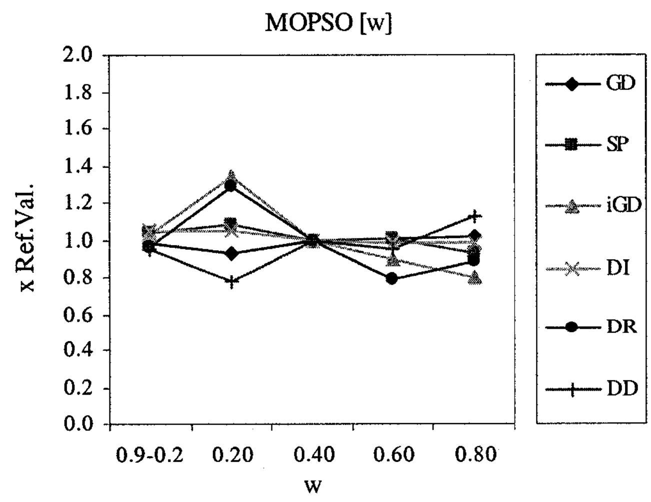



Figure B.3 – Sensitivity Analysis for Parameter *w* for Test Problem TD

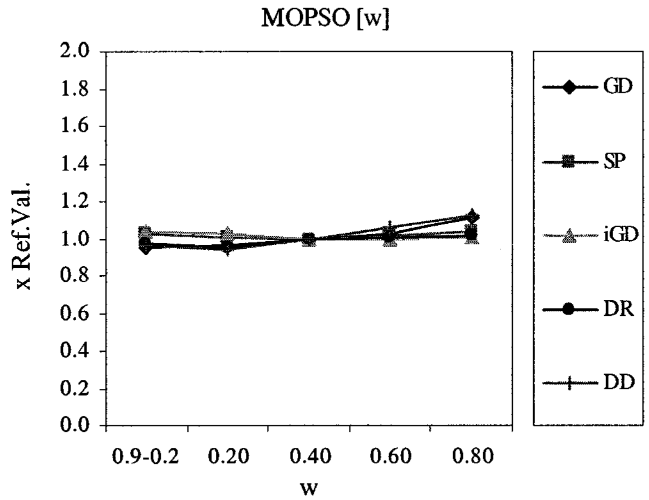



Figure B.4 – Sensitivity Analysis for Parameter $w$ for Test Problem WEF

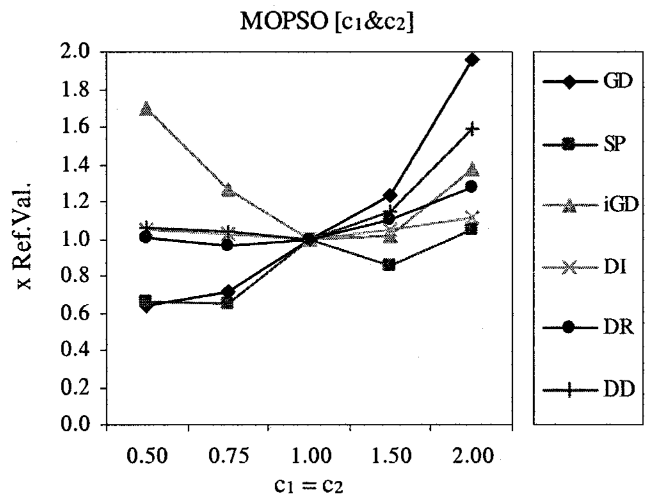



Figure B.5 – Sensitivity Analysis for Parameter $c_1$&$c_2$ for Test Problem 1

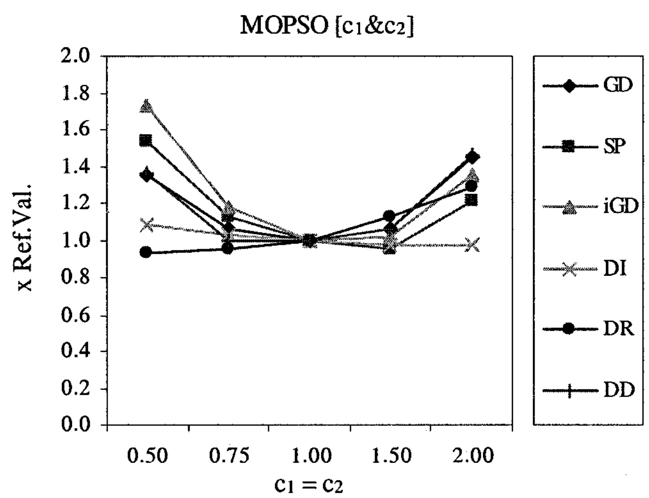



Figure B.6 – Sensitivity Analysis for Parameter $c_1$&$c_2$ for Test Problem 5

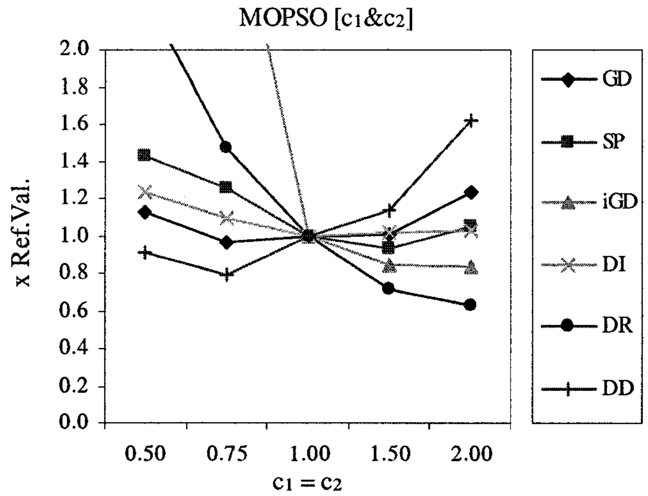



Figure B.7 – Sensitivity Analysis for Parameter $c_1\&c_2$ for Test Problem TD

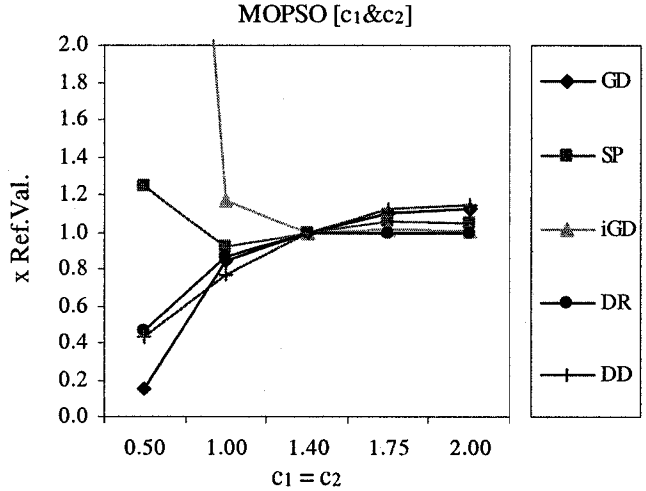



Figure B.8 – Sensitivity Analysis for Parameter $c_1\&c_2$ for Test Problem WEF

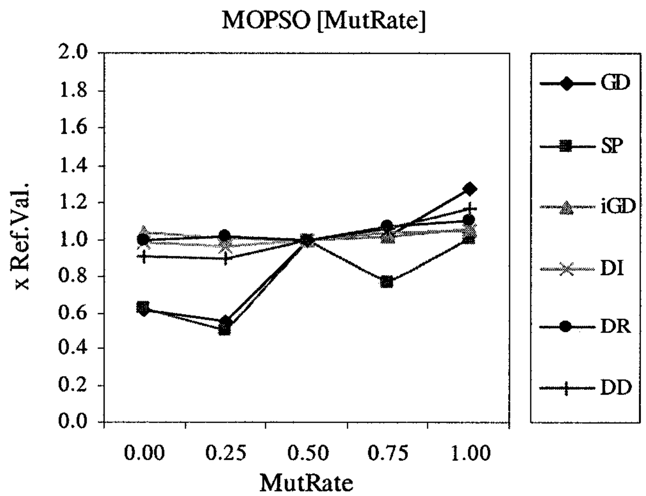



Figure B.9 – Sensitivity Analysis for Parameter *MutRate* for Test Problem 1

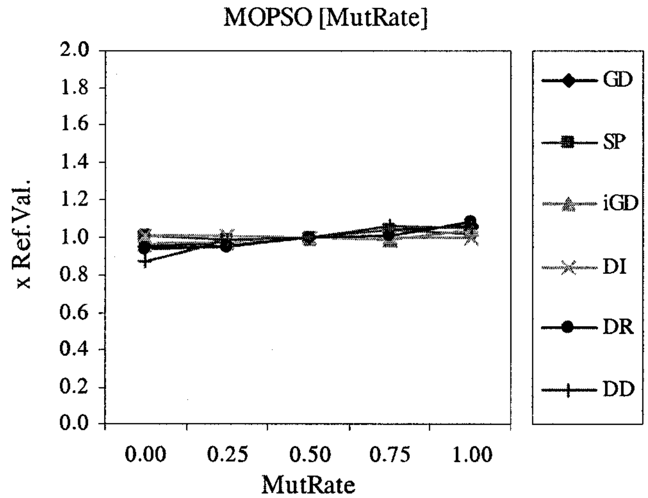



Figure B.10 – Sensitivity Analysis for Parameter *MutRate* for Test Problem 5

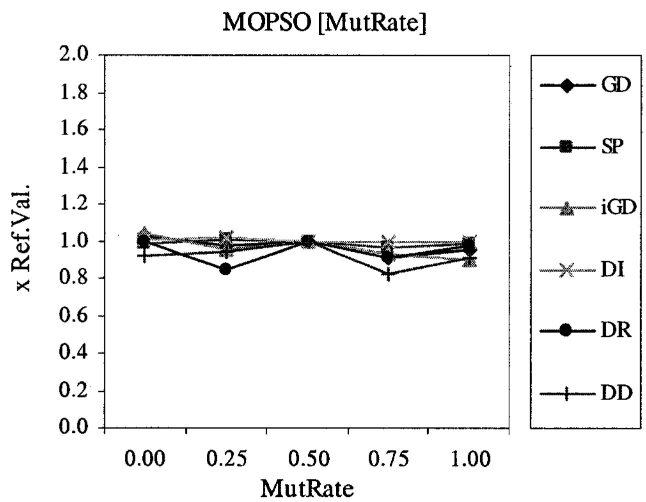



Figure B.11 – Sensitivity Analysis for Parameter *MutRate* for Test Problem TD

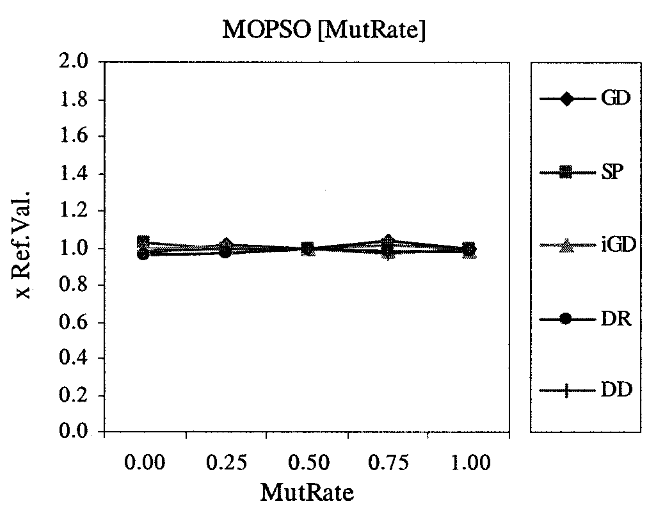



Figure B.12 – Sensitivity Analysis for Parameter *MutRate* for Test Problem WEF

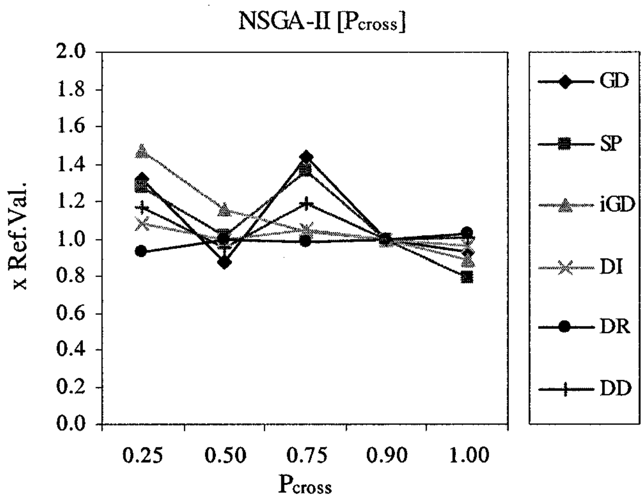



Figure B.13 – Sensitivity Analysis for Parameter $P_{cross}$ for Test Problem 1

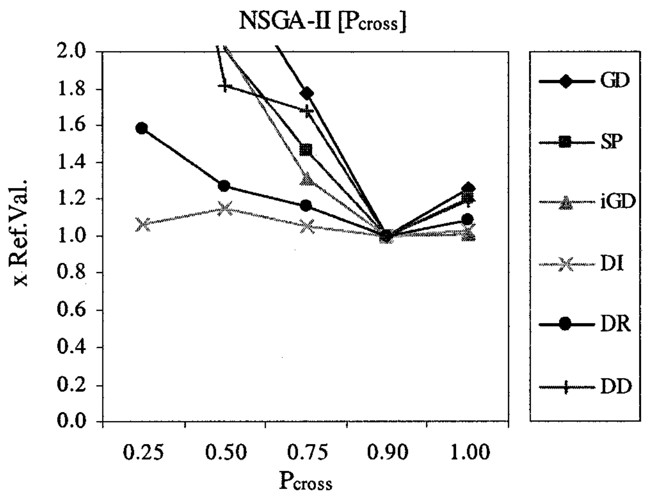



Figure B.14 – Sensitivity Analysis for Parameter $P_{cross}$ for Test Problem 5

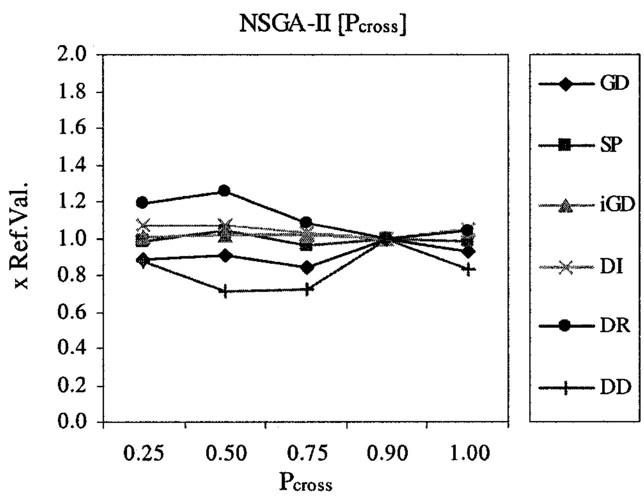



Figure B.15 – Sensitivity Analysis for Parameter $P_{cross}$ for Test Problem TD

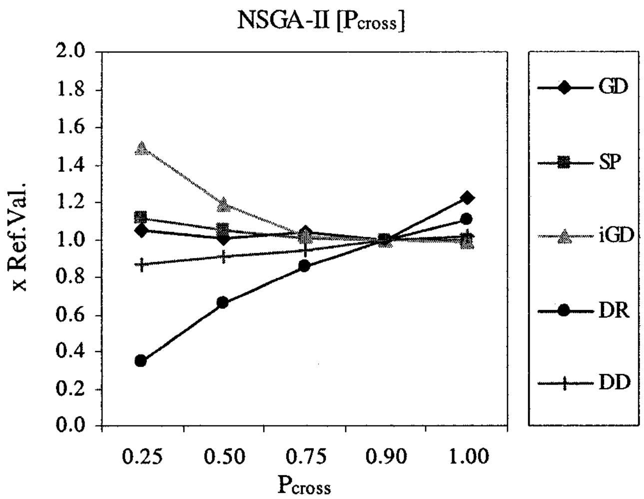



Figure B.16 – Sensitivity Analysis for Parameter $P_{cross}$ for Test Problem WEF

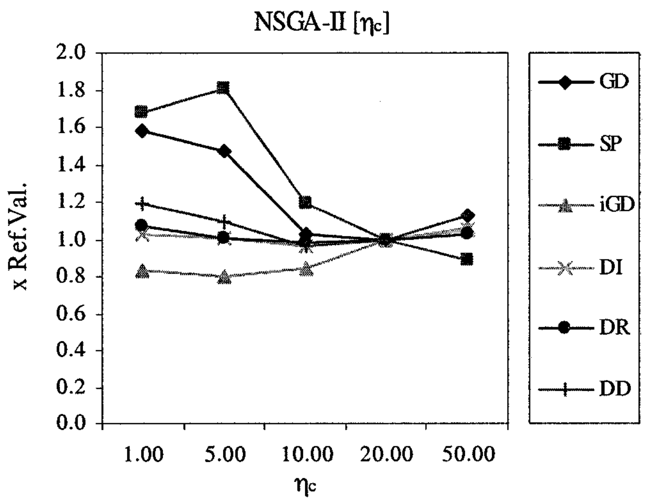



Figure B.17 – Sensitivity Analysis for Parameter $\eta_c$ for Test Problem 1

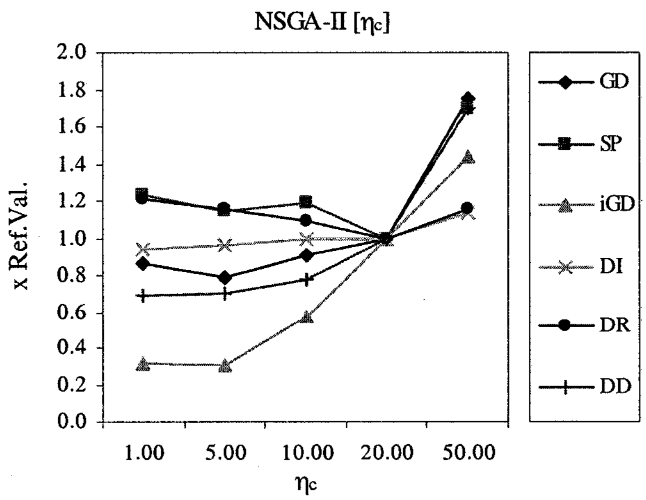



Figure B.18 – Sensitivity Analysis for Parameter $\eta_c$ for Test Problem 5

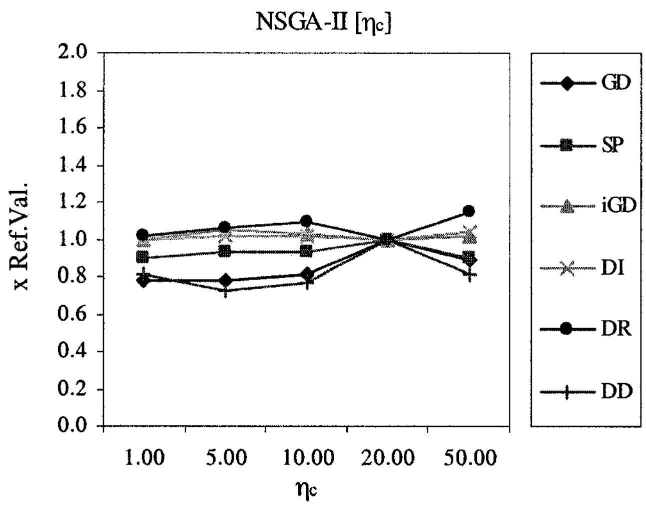



Figure B.19 – Sensitivity Analysis for Parameter $\eta_c$ for Test Problem TD

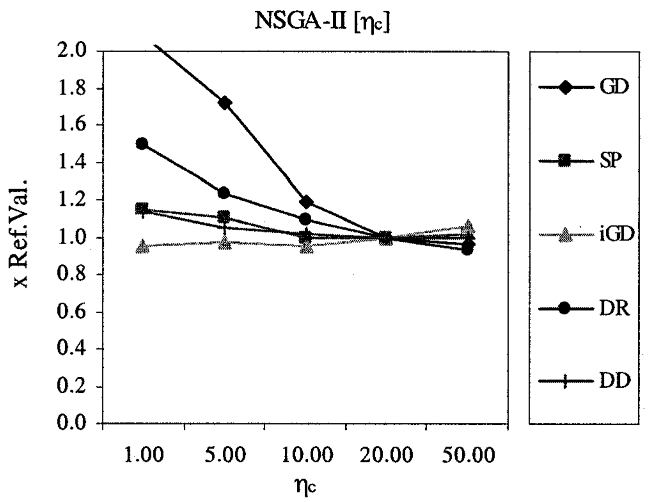



Figure B.20 – Sensitivity Analysis for Parameter $\eta_c$ for Test Problem WEF

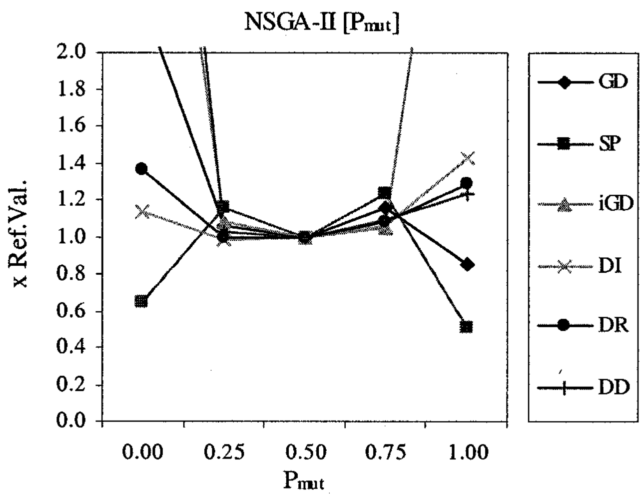



Figure B.21 – Sensitivity Analysis for Parameter $P_{mut}$ for Test Problem 1

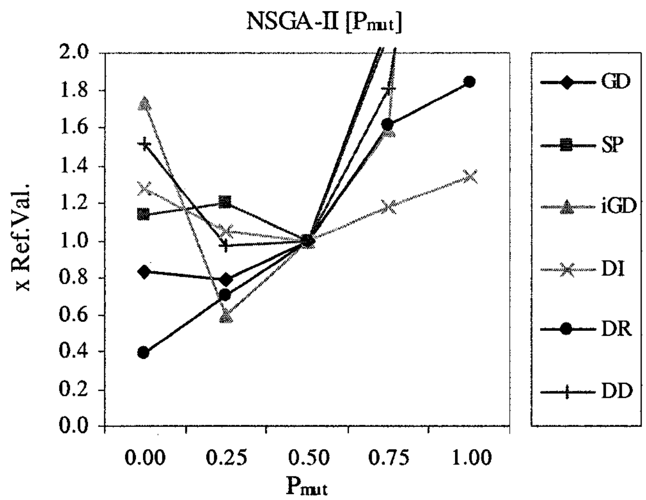



Figure B.22 – Sensitivity Analysis for Parameter $P_{mut}$ for Test Problem 5

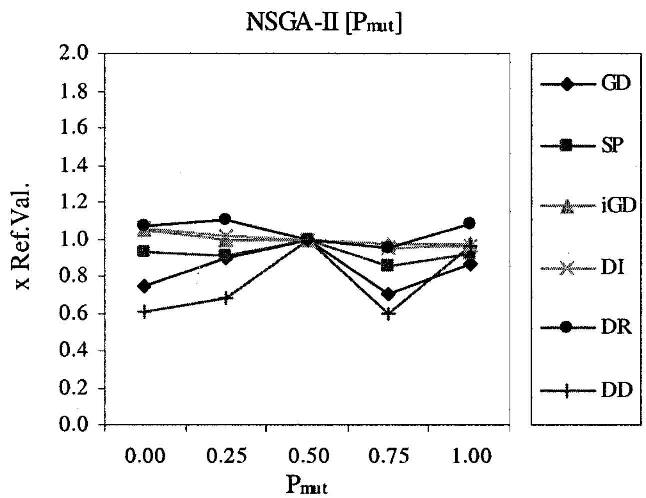



Figure B.23 – Sensitivity Analysis for Parameter $P_{mut}$ for Test Problem TD

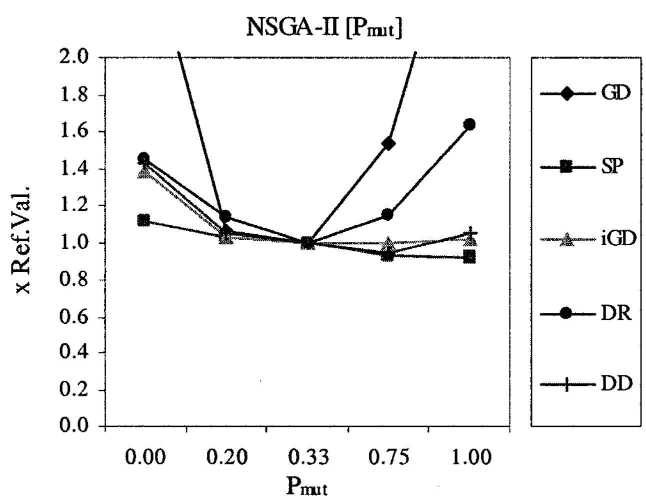



Figure B.24 – Sensitivity Analysis for Parameter $P_{mut}$ for Test Problem WEF

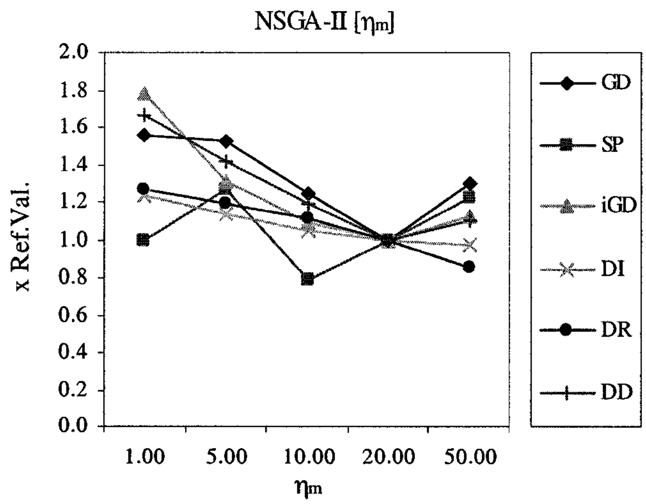



Figure B.25 – Sensitivity Analysis for Parameter $\eta_m$ for Test Problem 1

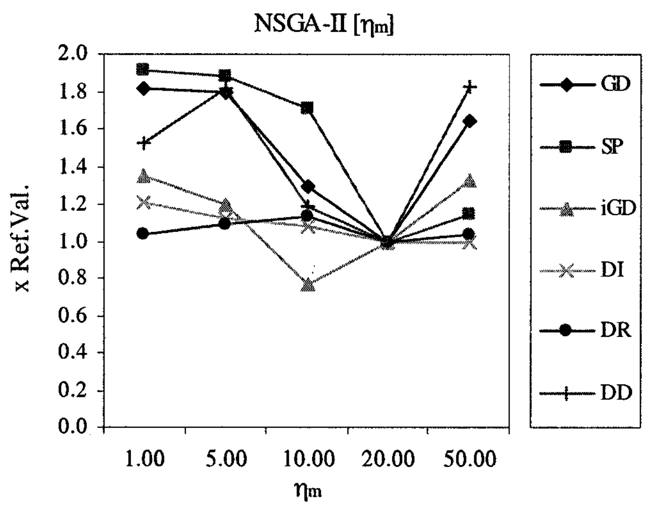



Figure B.26 – Sensitivity Analysis for Parameter $\eta_m$ for Test Problem 5

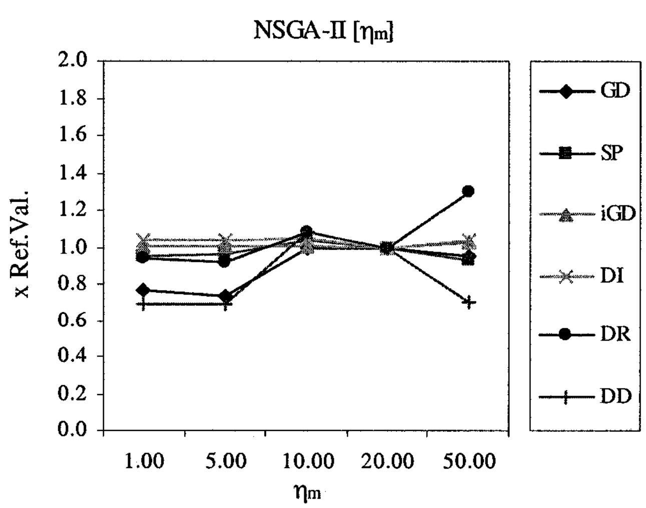



Figure B.27 – Sensitivity Analysis for Parameter $\eta_m$ for Test Problem TD

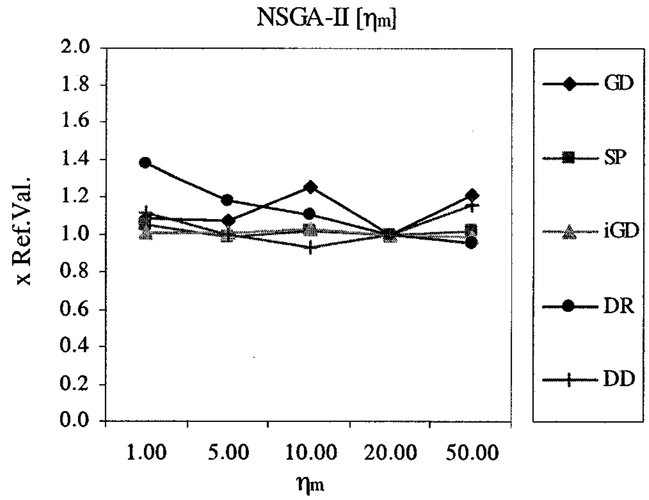



Figure B.28 – Sensitivity Analysis for Parameter $\eta_m$ for Test Problem WEF

# APPENDIX C: INTERFACES OF THE IMPLEMENTED SOLVERS

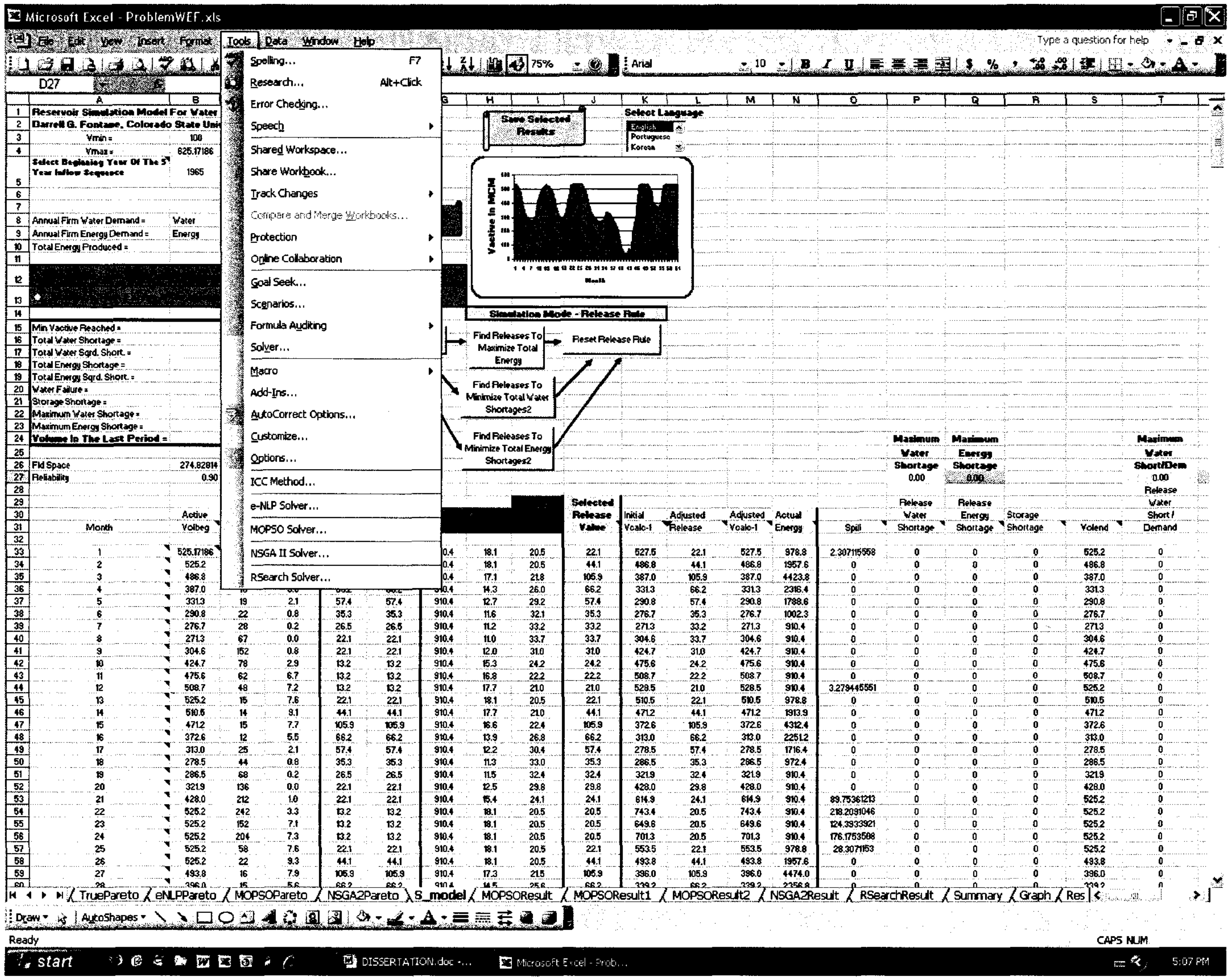



Figure C.1 – Microsoft Excel Tools Menu with the Links to the Various Solvers

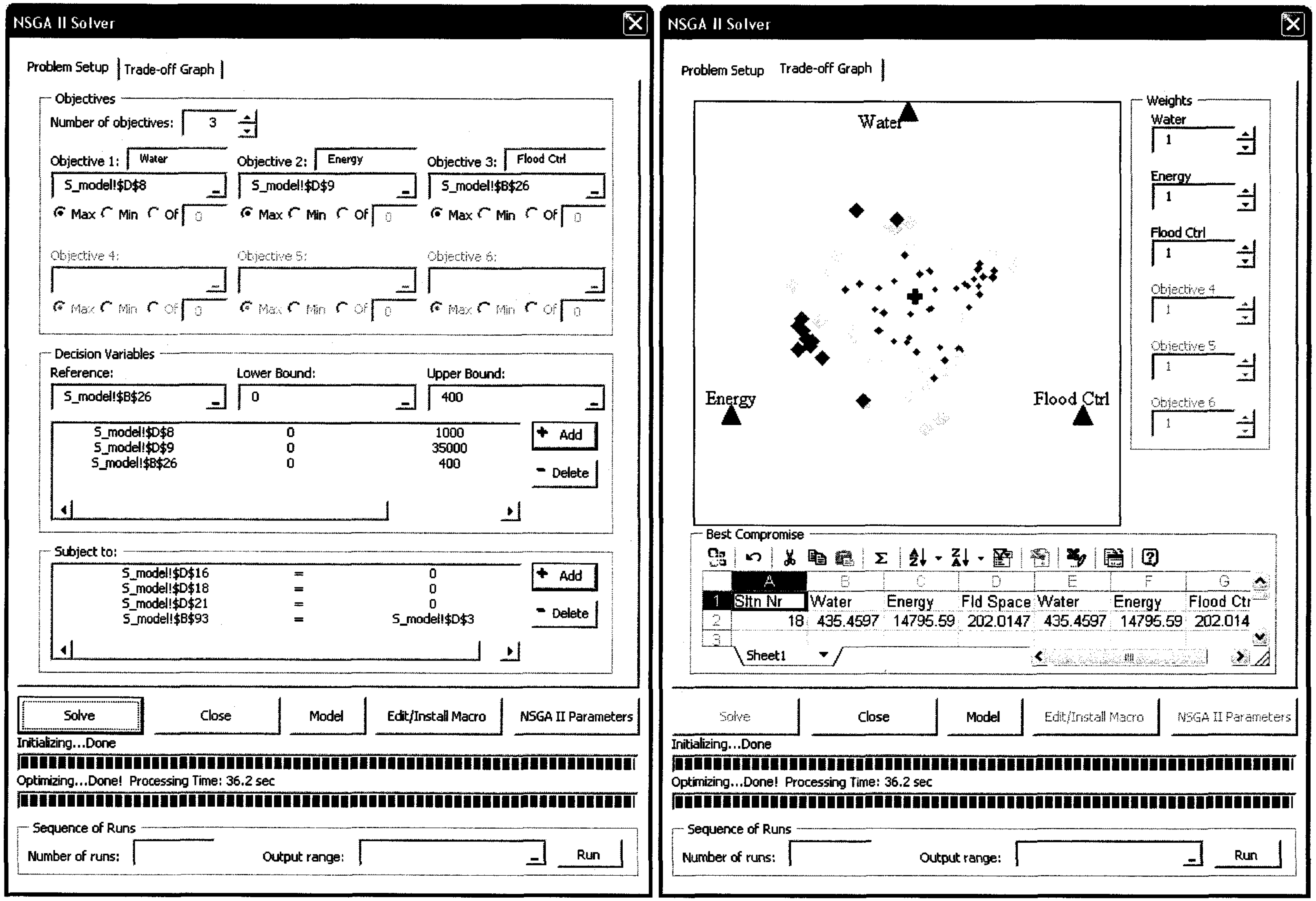



Figure C.2 – NSGA-II Solver Interface

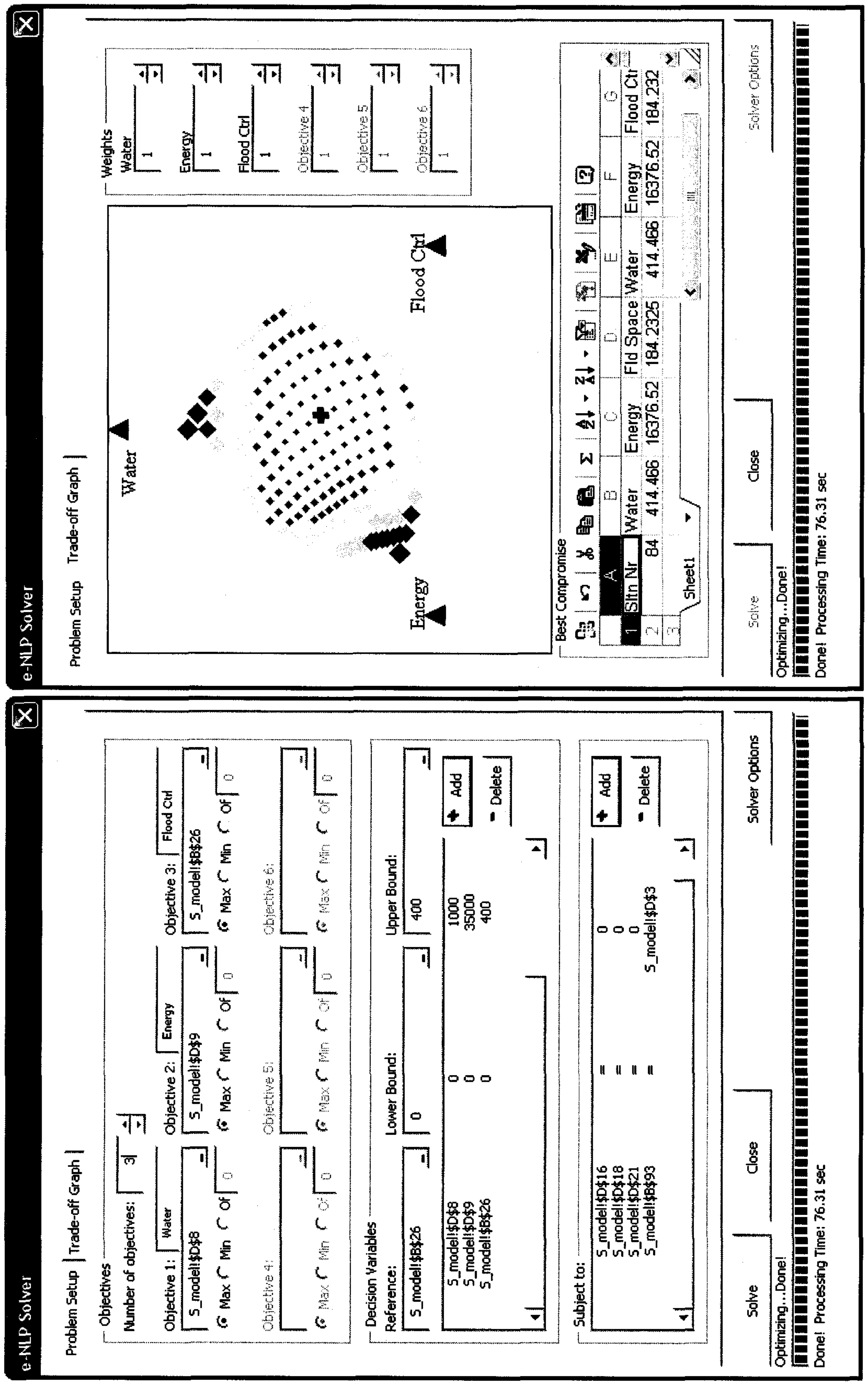



Figure C.3 – ε-NLP Solver Interface

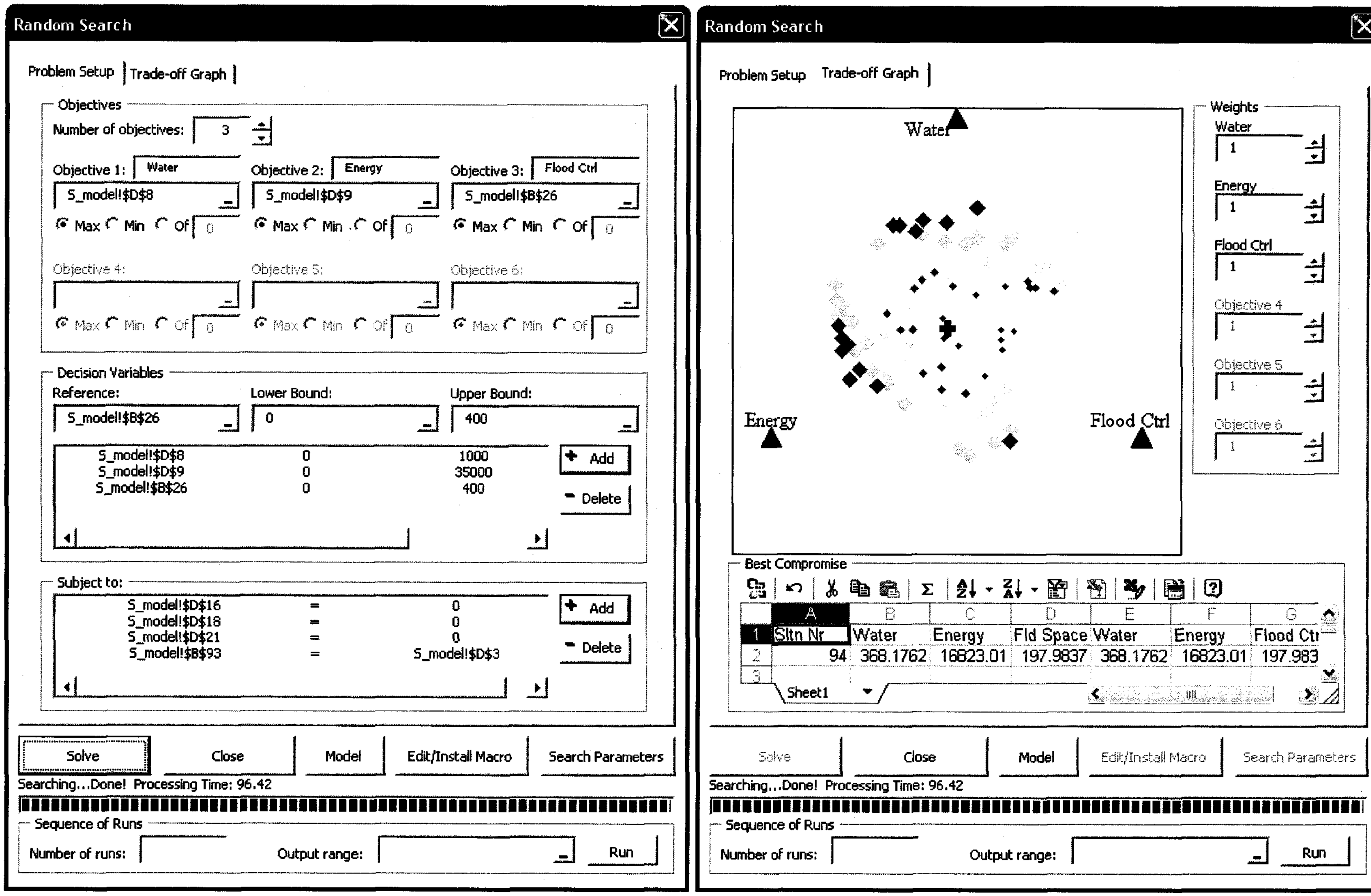



Figure C.4 – RSearch Solver Interface