This document is downloaded from DR-NTU (https://dr.ntu.edu.sg)
Nanyang Technological University, Singapore.

Memetic algorithm using multiple surrogates for complex engineering design optimization

Zhou, Zongzhao

2008

Zhou, Z. Z. (2008). Memetic algorithm using multiple surrogates for complex engineering design optimization. Doctoral thesis, Nanyang Technological University, Singapore.

https://hdl.handle.net/10356/13587

https://doi.org/10.32657/10356/13587

Downloaded on 26 Aug 2022 02:39:39 SGT

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Nanyang Technological University

Memetic Algorithm Using Multiple Surrogates for Complex Engineering Design Optimization

Zhou Zongzhao

School of Computer Engineering

A thesis submitted to Nanyang Technological University
in fulfillment of the requirements of
degree of Doctor of Philosophy

2008

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Abstract

Complex engineering design (CED) optimization problems in science and engineering commonly have large design spaces. In such design spaces, typically thousands of exact fitness evaluations are required to locate a near optimal design. Often in photonics, electromagnetic, aerospace, biomedical and microwave circuits detailed design processes, variable-fidelity analysis codes are employed to strike a balance between design cost, time and estimation accuracy. Nevertheless, in analysis and design optimization processes where high-fidelity analysis codes are used, each exact fitness evaluation requiring the simulation of analysis codes may cost hours of supercomputer time. Therefore, the overwhelming part of the total run time in CED optimization is usually taken up by the simulation of analysis codes. This often poses a serious impediment to the practical application of high-fidelity analysis codes and evolutionary algorithms to CED optimization problems.

In this dissertation work, the research focus has been placed on the use of multiple surrogate models in standard memetic algorithm (MA) to mitigate the costly CED optimization process. In this thesis, a novel hierarchical surrogate-assisted memetic algorithm (HSAMA) combining both global and local surrogate models for accelerating the optimization process is proposed and described. The performance of the proposed algorithm is analyzed by using a series of commonly used benchmark test functions. Furthermore, the proposed algorithm is also applied to aerodynamic shape design. Numerical results show that the HSAMA algorithm is capable of achieving good designs efficiently under a limited computational budget.

Further, the impact of uncertainty introduced by approximation errors, i.e., '*curse and blessing of uncertainty*', is illustrated and demonstrated on surrogate-assisted memetic algorithm (SAMA). Inspired by this finding, a novel multi-surrogates assisted memetic

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

algorithm (MSAMA) that uses multiple surrogate modeling techniques during the local search phase in MA is proposed to leverage from the effect of '*curse and blessing of uncertainty*' in evolutionary optimization. Empirical results obtained demonstrate that MSAMA significantly outperforms the standard GA, standard MA and existing SAMA variants with only one surrogate modeling technique when solving computationally expensive optimization problems. This leads to a new paradigm in surrogate-assisted evolutionary algorithm (SAEA) design.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Acknowledgments

I would like to express my sincere gratitude to my supervisor, Dr. **Ong Yew Soon**, for his intellectual guidance, constant encouragement and persistent patience throughout my Ph.D research. This thesis would not have been possible without his guidance and support.

I would also like to thank all my friends: **Zhu Ning, Zhang Yu, Zhou Zhi, Zhou Chen, Li Xiangjun, Wu Jianhua, Hu Yiqun, Wang Huiqing, Zou Zixuan, Zhu Zexuan, Dudy Lim, Ng Hee Khiang, Chen Lei, Chen Xi, Guo Jing, Zhang Yuquan, Huang Haijie, Xie Jun, Xie Jiawei, Chu Yang, Ye Minghua, Yi Haoran, Feng Zhou, Zhang Ke, Yu Kuan, Zhou Bin, Tao Dayong, Li Jiaming, Liu Xin, Dr Gao Deyun**, for their unhesitating help and precious friendship.

My utmost gratitude to **My Parents, My Sisters** and **My Wife, Xiao Rongmei,** for their constant support, understanding and love, even though I could never thank them enough.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Table of Contents

Abstract i
Acknowledgments iii
Table of Contents iv
List of Figures vii
List of Tables xi
Acronyms and Abbreviations xiii
Symbols xv

1 Introduction **1**
1.1 Research Background 1
1.2 Problem Statement 2
1.3 Research Objective and Focus 5
1.4 Organization of the Thesis 7

2 Literature Survey **8**
2.1 Evolutionary Algorithm 8
2.1.1 Genetic Algorithm 9
2.1.2 Memetic Algorithm 13
2.2 Surrogate-Assisted Evolutionary Algorithm 15
2.3 Surrogate Modeling Techniques in SAEA 20
2.3.1 Surrogate Model Selection in SAEA 22
2.3.2 Global vs Local Surrogate Model 23
2.4 Model Management Methods in SAEA 24
2.4.1 Individual-based Evolution Control 27
2.4.2 Generation-based Evolution Control 31

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

2.4.3 Memetic-Based Evolution Control 33
2.5 Summary 34

3 A Study on Surrogate-Assisted Memetic Algorithms 35
3.1 Introduction 35
3.2 Variants of Surrogate Assisted Memetic Algorithm 35
3.2.1 Standard SAMA 36
3.2.2 SAMA with Trust Region Framework 38
3.3 Local Search Strategy 38
3.3.1 Radial Basis Function 40
3.3.2 Trust Region Framework 42
3.4 Empirical Study on SAMA Variants 44
3.5 Summary 51

4 Hierarchical Surrogate-Assisted Memetic Algorithm 52
4.1 Outline of HSAMA Optimization Framework 52
4.2 Global Search Strategy 55
4.2.1 Data Parallel Gaussian Process Modeling 56
4.2.2 Polynomial Regression Modeling Technique 60
4.2.3 Pre-Selection Criterion 62
4.3 Performance Analysis of GP-HSAMA 63
4.4 Comparative Study of HSAMA Variants 70
4.4.1 Effect of Different Order of PR Model 70
4.4.2 Comparison of PR- and GP-HSAMAs 73
4.4.3 Wall Clock Time of PR- and GP-HSAMAs 74
4.5 Aerodynamic Shape Design Optimization 78
4.6 Summary 80

5 Multi-Surrogates Assisted Memetic Algorithm 82
5.1 Impact of Uncertainty 82
5.1.1 Impact of Uncertainty on SAMA 83
5.1.2 Empirical Study on Impact of Uncertainty 85

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

5.2 Multi-Surrogates Assisted Memetic Algorithm 95
5.2.1 Outline of MSAMA Algorithm 96
5.2.2 Empirical Study on MSAMA 99
5.3 Summary 111

6 Conclusions and Future Work **112**
6.1 Conclusions 112
6.2 Future Work 113
6.2.1 Pre-Selection Criterion in HSAMA 113
6.2.2 Enhancement on MSAMA 114
6.2.3 Analysis of Global Convergence Rates 114
6.2.4 Grid-enabled SAMA variants 115
6.2.5 Effect of the Genetic Operators 116
6.2.6 High Dimensionality Optimization 116

List of Publications **117**

Appendix A Benchmark Test Problems **118**
A.1 Step Test Function 118
A.2 Sphere Test Function 120
A.3 Rosenbrock Test Function 120
A.4 Ackley Test Function 121
A.5 Griewank Test Function 121
A.6 Rastrigin Test Function 123

Appendix B Aerodynamic Shape Design Optimization Problem **124**

Appendix C Lamarckian and Baldwinian Learning **128**

References **129**

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

List of Figures

1.1 A typical design cycle of complex engineering design problem. 2
1.2 A typical optimization workflow based on computer simulation. 2

2.1 One point crossover. 11
2.2 Two point crossover. 11
2.3 Uniform crossover. 12
2.4 Concept of building surrogate model for computer simulations. 21
2.5 An example of a false minimum in the surrogate model (From [71]). . . . 25

3.1 Work flow of the SAMA-TRF 39
3.2 Average fitness value among 20 independent runs for SAMA and SAMA-TRF with different mutation rate and crossover procedure on Ackley function. 45
3.3 Average fitness value among 20 independent runs for SAMA and SAMA-TRF with different mutation rate and crossover procedure on Griewank function. 46
3.4 Average fitness value among 20 independent runs for SAMA and SAMA-TRF with different mutation rate and crossover procedure on Rastrigin function. 47

4.1 Characteristics of the *PoI* criterion when the GP model is trained on points generated by a one-dimensional function. 64
4.2 Convergence trends of the GA, SAGA-GS, SAMA-TRF and GP-HSAMA framework for Sphere function. 68
4.3 Convergence trends of the GA, SAGA-GS, SAMA-TRF and GP-HSAMA framework for Rosenbrock function. 68

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

4.4 Convergence trends of the GA, SAGA-GS, SAMA-TRF and GP-HSAMA framework for Ackley function. 69
4.5 Convergence trends of the GA, SAGA-GS, SAMA-TRF and GP-HSAMA framework for Griewank function. 69
4.6 Convergence trends of the GA, SAGA-GS, SAMA-TRF and GP-HSAMA framework for Rastrigin function. 70
4.7 2-D Ackley function and its PR approximation models. 71
4.8 Averaged search trends of PR-HSAMA for different orders on 20-dimensional Ackley function. 72
4.9 Averaged search trends of PR-HSAMA for different orders on 20-dimensional Griewank function. 72
4.10 Averaged search trends of SAMA-TRF, GP-HSAMA, and PR-HSAMA on 20-dimensional Ackley function. 74
4.11 Averaged search trends of SAMA-TRF, GP-HSAMA, and PR-HSAMA on 20-dimensional Griewank function. 75
4.12 Wall clock time of GP-HSAMA and PR-HSAMA on the 20-Dimensional $100s$ Griewank Function. 76
4.13 Wall clock time of GP-HSAMA and PR-HSAMA on the 20-Dimensional $10s$ Griewank Function. 77
4.14 Convergence trends of the GA, SAMA-TRF and GP-HSAMA framework for aerodynamic shape design problem. 80

5.1 '*Curse of uncertainty*' in surrogate-assisted memetic algorithm 84
5.2 'Blessing of uncertainty' in surrogate-assisted memetic algorithm 86
5.3 Convergence trends of the GA, SAMA-PR, SAMA-RBF and SAMA-Perfect on 20D Sphere function 88
5.4 Convergence trends of the GA, MA, SAMA-PR, SAMA-RBF and SAMA-Perfect on 20D Step function 89
5.5 Convergence trends of the GA, MA, SAMA-PR, SAMA-RBF and SAMA-Perfect on 20D Ackley function 89
5.6 Convergence trends of the GA, MA, SAMA-PR, SAMA-RBF and SAMA-Perfect on 20D Griewank function 90

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

5.7 Convergence trends of the GA, MA, SAMA-PR, SAMA-RBF and SAMA-Perfect on 20D Rastrigin function 90
5.8 $t - \hat{t}$ plot of PR modeling for 20D Ackley function 93
5.9 $t - \hat{t}$ plot of RBF modeling for 20D Ackley function 94
5.10 $t - \hat{t}$ plot of PR modeling for 20D Rastrigin function 94
5.11 $t - \hat{t}$ plot of RBF modeling for 20D Rastrigin function 95
5.12 Work Flow of MSAMA 97
5.13 Convergence trends of the MSAMA variants on 20D Sphere function . . . 101
5.14 Convergence trends of the MSAMA variants on 20D Step function 101
5.15 Convergence trends of the MSAMA variants on 20D Ackley function . . . 102
5.16 Convergence trends of the MSAMA variants on 20D Griewank function . 102
5.17 Convergence trends of the MSAMA variants on 20D Rastrigin function . 103
5.18 Convergence trends of the MSAMA-PF on 20D Sphere function 105
5.19 Convergence trends of the MSAMA-PF on 20D Step function 105
5.20 Convergence trends of the MSAMA-PF on 20D Ackley function 106
5.21 Convergence trends of the MSAMA-PF on 20D Griewank function 106
5.22 Convergence trends of the MSAMA-PF on 20D Rastrigin function 107
5.23 SAMA-PR, SAMA-RBF and MSAMA-PF Box plot of the best fitness in the final generation for 20D Sphere function 107
5.24 SAMA-PR, SAMA-RBF and MSAMA-PF Box plot of the best fitness in the final generation for 20D Ackley function 108
5.25 SAMA-PR, SAMA-RBF and MSAMA-PF Box plot of the best fitness in the final generation for 20D Griewank function 108
5.26 SAMA-PR, SAMA-RBF and MSAMA-PF Box plot of the best fitness in the final generation for 20D Rastrigin function 109

6.1 Redundant individuals within the same basin of a local minimum 113
A.1 2-D Step benchmark test function 119
A.2 1-D Sphere benchmark test function 119
A.3 2-D Rosenbrock benchmark test function 120
A.4 2-D Ackley benchmark test function 122

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

A.5 1-D Griewank benchmark test function 122
A.6 2-D Rastrigin benchmark test function 123
B.1 Forces acting on an airplane and airfoil. 125
B.2 Airfoil geometry characterized using 24-parameter Hicks-Henne functions. 127

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

List of Tables

3.1 Radial Basis Kernels 41
3.2 Average fitness value among 20 independent runs for SAMA and SAMA-TRF with different mutation rates on all benchmark test functions. . . . 48
3.3 Best fitness value among 20 independent runs for SAMA and SAMA-TRF with different mutation rates on all benchmark test functions. 48
3.4 Ranking of average fitness value among 20 independent runs for SAMA and SAMA-TRF with different mutation rates on all benchmark test functions. 49
3.5 Ranking of best fitness value among 20 independent runs for SAMA and SAMA-TRF with different mutation rates on all benchmark test functions. 49
3.6 Average fitness value among 20 independent runs for SAMA and SAMA-TRF with different crossover procedure on all benchmark test functions . 50
3.7 Best fitness value among 20 independent runs for SAMA and SAMA-TRF with different crossover procedure on all benchmark test functions. 50
3.8 Ranking of average fitness value among 20 independent runs for SAMA and SAMA-TRF with different crossover procedures on all benchmark test functions. 50
3.9 Ranking of best fitness value among 20 independent runs for SAMA and SAMA-TRF with different crossover procedures on all benchmark test functions. 50

4.1 The configurations of GA, SAGA-GS, SAMA-TRF and HSAMA used in empirical study. 65
4.2 Average wall clock time (seconds) to construct surrogate model 76
4.3 The configurations of GA, SAMA-TRF, HSAMA for aerodynamic shape design 79

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

5.1 The $rmse$ and r correlation measures for PR and RBF surrogates 92
5.2 The configurations of SAMA and MSAMA used in empirical study. . . . 99
5.3 Abbreviation of SAMA and MSAMA Variants 100
5.4 Criteria for measuring performance. 109
5.5 Statistics of SAMA-PR, SAMA-RBF and MSAMA-PF for Sphere Function.110
5.6 Statistics of SAMA-PR, SAMA-RBF and MSAMA-PF for Ackley Function.110
5.7 Statistics of SAMA-PR, SAMA-RBF and MSAMA-PF for Griewank Function. 110
5.8 Statistics of SAMA-PR, SAMA-RBF and MSAMA-PF for Rastrigin Function. 110

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Acronyms and Abbreviations

AOA ..... Angle Of Attack
CED ..... Complex Engineering Design
CFD ..... Computational Fluid Dynamics
DACE ..... Design and Analysis of Computer Experiments
DPGP ..... Data Parallel Gaussian Process
EA ..... Evolutionary Algorithm
FSQP ..... Feasible Sequential Quadratic Programming
GA ..... Genetic Algorithm
GA-IPE ..... GA with Inexact Pre-Evaluation phase
GP ..... Gaussian Process
GP-HSAMA ..... GP global surrogate model based HSAMA
HSAMA ..... Hierarchical Surrogate-Assisted Memetic Algorithm
MA ..... Memetic Algorithm
MMP ..... Mean of Model Prediction
RBF ..... Radial Basis Functions
SAGA-GS ..... Surrogate-Assisted Genetic Algorithm with Global-search Strategy
SAMA ..... Surrogate-Assisted Memetic Algorithm
SAMA-PR ..... SAMA with PR model
SAMA-Perfect ..... SAMA with Perfect model
SAMA-RBF ..... SAMA with RBF model
SAMA-TRF ..... SAMA with Trust Region Framework
MSAMA ..... Multi-Surrogates Assisted Memetic Algorithm
MSAMA-EP ..... MSAMA with Perfect and PR models
MSAMA-EF ..... MSAMA with Perfect and RBF models

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

MSAMA-PF ..... MSAMA with PR and RBF models
SAEA ..... Surrogate-Assisted Evolutionary Algorithm
PR ..... Polynomial Regression
PoI ..... Probability of Improvement
TRF ..... Trust Region Framework

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Symbols

d ..... number of dimension
$\mathcal{D}$ ..... training dataset
$e(\mathbf{x}_i)$ ..... approximation error of design point $\mathbf{x}_i$
E ..... least square error
f_M ..... merit function
$f(\mathbf{x})$ ..... exact objective (fitness) function
$\hat{f}(\mathbf{x})$ ..... surrogate model for exact fitness function
$g(\mathbf{x})$ ..... constrain function
$\hat{g}(\mathbf{x})$ ..... surrogate model for constrain function
k_{max} ..... max iteration number of local search
m ..... order number of PR model
M ..... number of the nearest neighbors
n ..... variables size
t ..... output fitness variable
$\hat{t}$ ..... approximated output fitness variable
T_{obj} ..... time of calling a single objective function
$T_{surrogate}$ ..... time of building a surrogate model
$\mathbf{x}$ ..... input design variable
$\mathbf{x}_l$ ..... lower bound of input design
$\mathbf{x}_u$ ..... upper bound of input design
ρ ..... merit for updating the trust-region radius
Ω ..... trust-region radius
σ ..... standard deviation

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Chapter 1

Introduction

1.1 Research Background

Over the last decades, the process of design in engineering and science has been transformed by the introduction of massive computing power and advances in Internet technology, computational sciences and intelligence. There has been a move away from paper-based systems towards computer modeling and simulations that involve the use of highly sophisticated computer-aided design packages, high-fidelity analysis packages, meshing packages, complex optimization tools and others.

The move has led us to the development of increasingly advanced products such as the space telescope, large airliners and space shuttles. Nevertheless, this recent move towards computer modeling and simulations bring with it the need to access more complex and specialized software libraries, massive or rare data sets, and the unique contributions of the design teams in other disciplines, locations, and organizations. Those engaged in design research are focusing on these issues and how they can be taken forward.

A typical design cycle of the complex engineering design (CED) is shown in Figure 1.1. The design cycle begins with the computer aided design (CAD) modeling, followed by pre-processing, optimization and visualization phases. Engineers often use CAD tools to perform the design and evaluate this via numerical simulations. He/She then enters a design-evaluate-redesign process until the design is satisfied based on his/her knowledge

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

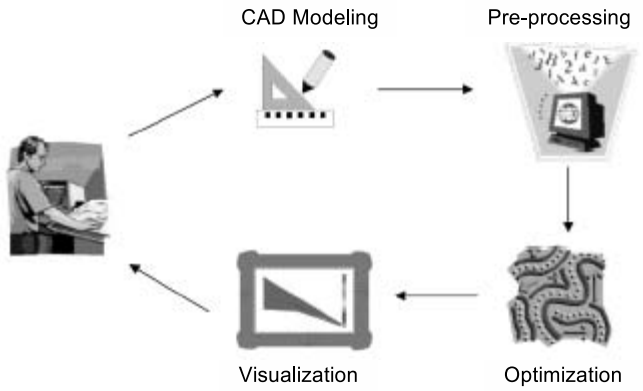


Figure 1.1: A typical design cycle of complex engineering design problem.

and experiences. In the CED life cycle, optimization represents one of the most crucial phase since it takes up most of the entire design time.

1.2 Problem Statement

Optimization is a mature technology that has been studied extensively by researchers over the last decades. It aims to find a set of values for the variables that minimizes (or maximizes) the objective function whilst respecting the constraints.

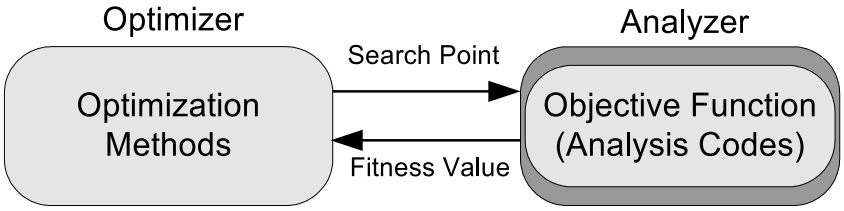


Figure 1.2: A typical optimization workflow based on computer simulation.

Figure 1.2 shows a typical optimization workflow for optimizing problem in the form of computer simulation. This simple workflow is made up of two major components: optimizer and analyzer. The optimizer serves as the guide for the search while the

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

analyzer provides the figure of merits of each potential design. The search proceeds repeatedly according to the optimization strategy employed until the stopping criterion is met.

Often in CED optimization process such as photonics, electromagnetic, aerospace, biomedical, micro-electro mechanical system, microwave circuits and coupled multidisciplinary systems detailed design processes, variable-fidelity analysis codes are employed to strike a balance between design cost, time and estimation accuracy. Nevertheless, in analysis and design optimization processes where high-fidelity analysis codes are used, each objective function evaluation requires the simulation of the high-fidelity analysis codes, such as finite element analysis, computational structural mechanics, computational fluid dynamics or computational electro magnetics etc., may cost hours of supercomputer time. Therefore, the overwhelming part of the total run time in CED optimization is usually taken up by the evaluations of the objective function using the high-fidelity analysis codes.

Applying optimization techniques in the engineering design process helps improve the performance and quality of existing products and potentially leads to novel designs, which can be crucial in maintaining competitiveness in world markets. There are several optimization techniques which are now available and used in the engineering optimization community.

Conventional numerical optimization techniques are commonly used methods. Typical conventional numerical methods are gradient-based methods, steepest-descent methods, conjugate-gradient, quadratic programming, pattern search methods and linear approximation methods [1], [2], [3]. These methods usually work well in problems with low dimension and smooth search spaces. In order to apply these methods successfully, a relatively good starting point must be carefully selected. The search improvement over the starting point is usually a local optimum since the local search solver is likely to move

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

to the nearest local optimum or stop if it encounters a numerical hurdle. Conventional numerical optimization methods have the known advantage of their efficiency; however, they are very sensitive to starting point selection and are very likely to stop at local optima, hence, they are often referred as local search methods.

To increase the robustness and probability of obtaining a globally improved design, stochastic optimization methods, a form of global search methods, are often adopted as the principal algorithms in the engineering design search process. Some of the most popular modern stochastic optimization methods including simulated annealing, tabu search and evolutionary algorithms (EAs). In particular, EAs have shown considerable success in locating the global optimum solution of optimization problems that may often be characterized by non-convex and disjoint solution spaces.

Unfortunately, when applying EAs for solving CED optimization problem, it remains a difficult task due to the long design cycle time. It is very time-consuming when high-fidelity analysis codes are involved where each evaluation of objective function may cost hours of supercomputer time. Meanwhile, typical EAs require thousands of objective function evaluations to locate a near optimal solution. This poses a serious impediment to the practical application of combining high-fidelity analysis codes and evolutionary optimization algorithms in CED. As a result, it is important to retain the appeal of evolutionary optimization algorithms and tackle the issue of computationally expensive design problem if high quality design is desired within tractable design time schedule.

Several emerging technologies may be used to make the use of EAs feasible for CED optimization. One promising way to significantly reduce the computational cost of EAs is using surrogate modeling techniques which employ computationally cheap surrogate model, often also referred as metamodel or approximation model, in place of computationally expensive fitness evaluation during evolutionary optimization [4], [5], [6], [7], [8], [9], [10], [11], and etc. Knowledge of past evaluations during evolutionary optimization

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

can be used to build empirical surrogate models that approximate the exact objective function. The surrogate model is then used to predict promising new solutions at a smaller evaluation cost than the original problem and guide the search for promising solutions. By introducing surrogate models, the computational burden can be greatly reduced since the efforts involved in building the surrogate model and optimization using it are much smaller than the standard approach of direct coupling the expensive analysis codes with the optimizer.

The other promising way to handle this problem is using Grid computing technology [12], [13], [14]. Grid computing aims to bring together large numbers of vastly distributed processing and storage capabilities across multiple control domains to solve particular problems in a secure, coordinated manner. Meanwhile, a well-known strength of EAs is their ability to partition the population of individuals among multiple compute nodes. Since the design optimization cycle time is directly proportional to the number of calls to the high-cost analysis codes, an intuitive way to reduce the total search time of evolutionary optimization algorithms is to parallelize the analysis of the design points. All design points within a single EA population may be evaluated simultaneously across multiple compute nodes. The benefits of combining the emerging Grid computing technology in the context of evolutionary design optimization are numerous. In particular, the Grid provides the infrastructure to facilitate distributed computing and parallelism by tapping on vast compute power and a secure means of solving large-scale optimization problems.

1.3 Research Objective and Focus

The CED optimization process is often constrained by requirements such as delivery within a fixed time/budget, or achieving a target performance. In this thesis, the research

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

objective is to mitigate the computationally expensive CED optimization process and deliver a better design within a restricted time scale.

In particular, a general nonlinear programming problem is considered:

$$\begin{aligned} \textbf{Minimize}: \quad & f(\mathbf{x}) \\ \textbf{Subject to}: \quad & g_i(\mathbf{x}) \leq 0, i = 1, 2, \ldots, p \\ & \mathbf{x}_l \leq \mathbf{x} \leq \mathbf{x}_u \end{aligned} \tag{1.1}$$

where $\mathbf{x}$ is the input design variable, $f(\mathbf{x})$ is the exact objective (fitness) function, $g(\mathbf{x})$ is the constrain function, $\mathbf{x}_l$ and $\mathbf{x}_u$ are lower and upper bound of input design, and p is the number of inequality constraints respectively.

In this research, the focus is on cases where the evaluation of $f(\mathbf{x})$ and $g(\mathbf{x})$ are computationally expensive, and it is desired to obtain a near optimal solution under a tractable computational budget through the use of surrogate approximation models. To be precise, if T_{obj} represents the computational effort for performing a single objective function call to the high-fidelity analysis code and $T_{surrogate}$ is the computational effort incurred in building the surrogate approximation model, an optimization problem is regarded to be computationally expensive if the following condition holds:

$$T_{obj} \gg T_{surrogate} \tag{1.2}$$

In the present research work, a literature survey on existing surrogate-assisted evolutionary algorithm (SAEA) optimization frameworks for solving engineering design optimization problems has been made. While the use of global or local surrogate model in SAEA has been studied in the literature, the use of multiple and diverse surrogate models in the SAEA remains to be a relatively unexplored research area. Further, the SAMA that employs surrogate models in either global/local search phase or both phases

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

of memetic algorithm (MA) represents a promising approach for accelerating the evolutionary search performance. Taking these cues, the research focus of this thesis is placed on designing novel SAEAs by using multiple surrogate modeling techniques in memetic algorithm for solving computationally expensive CED optimization problem under a tractable computational budget.

1.4 Organization of the Thesis

The remainder of this thesis is organized as follows:

In chapter 2, a literature review on surrogate modeling techniques and existing model management methods in SAEAs, for solving complex engineering design optimization problem is presented.

Chapter 3 investigates and compares the effects of genetic operators on two SAMA variants, i.e., the standard SAMA and SAMA-TRF.

Chapter 4 presents the proposed HSAMA optimization framework which employs both global and local surrogate models for accelerating evolutionary search process. Results obtained from numerical studies on a series of benchmark test functions and an aerodynamic airfoil problem are also presented and discussed in the chapter.

Chapter 5 presents and demonstrates the notion of '*curse and blessing of uncertainty*'. Further, the proposed MSAMA optimization framework that uses multiple dissimilar surrogate model techniques in the local search phase, so as to leverage from the effects of '*curse and blessing of uncertainty*' in surrogate models for solving computationally expensive optimization problem, is also presented.

Chapter 6 gives the major conclusions of this thesis and outlines some directions for the future research.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Chapter 2

Literature Survey

In this chapter, a brief overview of EAs, in particular GA and MA, is presented. Subsequently, the use of surrogate model in EA, i.e., surrogate-assisted evolutionary algorithm (SAEA), is introduced. Finally, a literature review on the surrogate modeling techniques and existing model management methods in SAEAs, for solving complex engineering design optimization problem is presented.

2.1 Evolutionary Algorithm

Evolutionary algorithms including genetic algorithms, evolutionary programming and evolution strategies are stochastic search methods that have been applied successfully for solving complex engineering optimization problems where classical deterministic methods are known to have failed [15].

The outline of a standard EA is shown in PseudoCode 1. The standard EA proceeds in an iterative manner by generating new populations $P[gen]$ of individuals from the old ones ($gen = 0, 1, 2, ...$). Every individual in the population is the encoded (binary, real, ...) version of a tentative solution. An evaluation of objective function associates a fitness value to every individual indicating its suitability to the problem. The canonical algorithm applies stochastic operators such as selection, crossover, and mutation on an initially random population in order to compute a whole generation of new individuals. In

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

a general formulation, variation operators are applied to create a temporary population $P'[gen]$, evaluate the resulting individuals, and get a new population $P[gen+1]$ by either using $P'[gen]$ or, optionally, $P[gen]$. The halting condition is usually set as reaching a pre-defined number of iterations of the algorithm, or to find an individual with a given error if the optimum, or computational budget is exhausted.

PseudoCode 1 Outline of a standard Evolutionary Algorithm

```
gen := 0; /* Initialize the generation counter. */
Initialize and evaluate P[gen]; /* Create an initial population. */
while (not stop condition, such as the population is not converged.) do
  P'[gen] := variation(P[gen]); /* Apply variation operators. */
  Evaluate P'[gen];
  Evaluate P[gen + 1] := select(P'[gen], P[gen]); /* Create an new population. */
  gen := gen + 1; /* Increase the generation counter. */
end while
```

Often, the fields of evolutionary computing, neural networks, and fuzzy logic, are listed together as techniques for solving problems by using numeric knowledge representation. Unlike conventional (hard) computing, they are tolerant of imprecision, uncertainty, partial truth and approximation. These features make them less brittle than standard approaches and, as a consequence, they offer adaptivity. This broader research field is known as *soft computing*.

2.1.1 Genetic Algorithm

As a breed of EAs, GA [16] is used as the major optimization framework in the research work. There are some good reasons for the widespread use of GA in various application domains. The most important among them are:

(i) it is robust and can capture the global optimal solution without being trapped to local optima.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

(ii) it may readily incorporate any existing analysis simulation solvers, such as finite element analysis, computational fluid dynamics, etc., with the minimum effort.

(iii) it only requires a fitness function measuring the fitness score of each individual.

(iv) it can easily be parallelized.

The GA is an iterative search algorithm based on an analogy with the process of natural selection of Darwinian theory and evolutionary genetics. The GA search aims to optimize a user-defined fitness function. To perform this task, GA maintains a population of candidate individuals, over the entire search space. At each or generation, a new population is created. This new generation generally consists of individuals which fit better than the previous ones into the external environment as represented by the fitness function. As the population iterates through successive generations, the individual will in general end toward the global optimum of the fitness function. To generate a new population on the basis of a previous one, GA performs three main steps: a) it evaluates the fitness score of each individual of the old population. b) it selects individuals on the basis of their fitness score, and c) it recombines these selected individuals using genetic operators. The typical genetic operators are crossover and mutation. Next the most basic forms of these operators are introduced.

2.1.1.1 Crossover

This operator randomly chooses a locus and exchanges the subsequences before and after that locus between two individuals to create two offspring. The two offspring are constructed from parent strings by exchanging parts of their individuals. The pair of individuals are selected to undergo crossover with probability p_c. A random number r_c is generated in the range between 0 and 1, and the individuals undergo crossover if $r_c \leq p_c$, otherwise the pair proceed without crossover. Typical values of p_c are 0.4 to 0.9. The following are some typical binary crossover operators:

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

(i) **One-point crossover**: Single point crossover [17] is the simplest form. A single crossover position is chosen at random and portions of two parents after the crossover position are exchanged to create two new offsprings. In other words, a point i is chosen randomly along the string length L. Two new strings are generated by swapping all bits between positions $i+1$ and L (see figure 2.1).

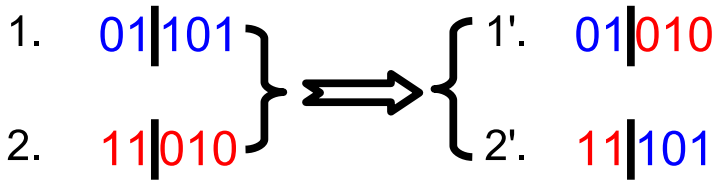


Figure 2.1: One point crossover.

(ii) **Two-point crossover**: Two cut points are chosen at random and the part of the parent chromosomes between these points is exchanged (see figure 2.2). Generally, two point crossover is better than one-point because it minimizes schema disruption better. In contrast, multi-point crossover operator in which multiple positions are chosen at random and the segments between them are exchanged, may not be useful because of extreme disruption to schemata.

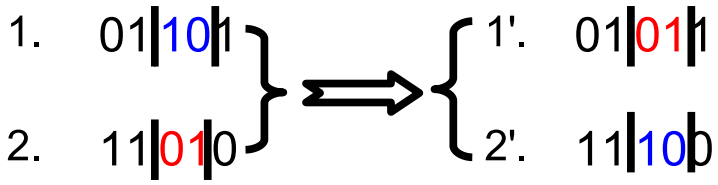


Figure 2.2: Two point crossover.

(iii) **Uniform crossover**: For each bit position 1 to L, randomly pick each bit from either of the two parent strings (see figure 2.3). This means that each bit is inherited independently from any other bit. As a result, uniform crossover is unbiased with

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

respect to length L. Generally, uniform crossover is considered as most disruptive. However, its disruptive character might be helpful in overcoming local optima for small populations.

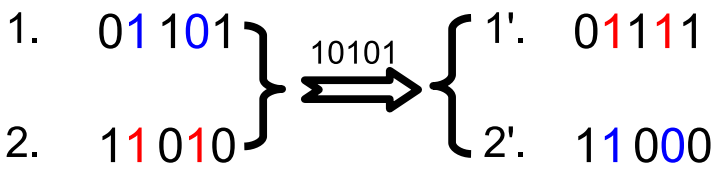


Figure 2.3: Uniform crossover.

2.1.1.2 Mutation

This operator randomly flips some of the bits in a chromosome. Mutation in nature introduces new variations and diversity thus giving rise to biological evolution. The motivation in the use of mutation is the same for crossover. There is a tradeoff between exploration of the search space by changing crossover point and mutation rate, and exploitation of the points already found. Too high mutation rates for the exploration purpose will slow down the convergence speed of the algorithm. On the other hand, too low rates will quickly lead to premature convergence to local optimal value. Normally, mutation should be used at very low levels. The probability of mutation p_m is typical of the order 0.001, i.e. one bit in every thousand will be mutated. The correct setting for p_m is problem independent. Many researchers have used $p_m = 1/L$, others $p_m = 1/(N\sqrt{L})$, where N is the population size.

If the GA has been correctly implemented, the population will evolve over successive generations so that the fitness of the best and the average individual in each generation increases towards the global optimum. Convergence is the progression towards increasing uniformity. A gene is said to have converged when 95% of the population share the same value. The population is said to have converged when all of the genes have converged.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

2.1.2 Memetic Algorithm

It is well-known that the EAs are capable of exploring and exploiting promising regions of the search space. They can, however, take a relatively long time to locate the exact local optimum in a region of convergence (and may sometimes not find the optimum with a sufficient precision). Torn and Zilinskas [18] observe that two competing goals, i.e., exploration and exploitation, govern the design of global search methods. Exploration is important to ensure global reliability; i.e., every part of the domain is searched enough to provide a reliable estimate of the global optimum; exploitation is also important since it concentrates the search effort around the best solutions found so far by searching their neighborhoods to produce better solutions.

Several search algorithms [19], [20], achieve these two goals using a combination of dedicated global and local searches. These are commonly known as hybrid methods. Hybrid evolutionary algorithm-local search methods, which incorporate local improvement procedures with traditional EAs may thus be used to improve the performance of EAs in search. Such hybrids have been used successfully to solve a wide variety of engineering design problems and experimental studies show that GAs hybrids not only often find better solutions than simple GAs, but also that they may search more efficiently [21], [18].

In this work, the core focus is on evolutionary algorithm where local search plays a significant role throughout the search, hence the term memetic algorithm (MA) [22] is used. Basically, the MA is a population-based heuristic search approach for optimization problems based on cultural evolution. They are inspired by Dawkins' notion of a meme [23] defined as a unit of information that reproduces itself while people exchange ideas. Hence, MA is a marriage between a population-based global search and the heuristic local search or meme made by each of the individuals. This makes perfect sense as the

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

combination of GA and local searches help facilitate exploration and exploitation in the search.

Two basic learning strategies are typically considered in MAs, i.e, Lamarckian and Baldwinian learning [24]. Lamarckian learning forces the genotype to reflect the result of improvement through placing the locally improved individual back into the population to compete for reproductive opportunities. In Baldwinian learning, the improvement procedures are only used to change the fitness landscape, but the solution that is found is not encoded back into the genetic string. The reader is referred to Appendix C for greater details.

PseudoCode 2 Outline of a standard MA

```
gen := 0; /* Initialize the generation counter. */
Create an initial population P[gen].
while (computational budget is not exhausted) do
  Evaluate P[gen] using the exact fitness function f(x).
  for (each non-duplicated individual in P[gen]) do
    Apply local searches using f(x) for local improvement.
    if (Lamarckian learning) then
      Replace both genetic string and fitness value of the original individual with
      those of the locally improved solution.
    else if (Baldwinian learning) then
      Replace the fitness value of the original individual with that of the locally im-
      proved solution.
    end if
  end for
  Apply standard EA operators to create a new population P[gen + 1].
  gen := gen + 1; /* Increase the generation counter. */
end while
```

A standard MA based on Lamarckian or Baldwinian learning is depicted in PseudoCode 2. A brief description of the memetic algorithm can be outlined as follows: Given a representation of an optimization problem, a certain number of individuals are created. The state of these individuals can be randomly chosen or according to a certain initialization procedure. After that, a local search is used for local improvement based on Lamarckian

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

or Baldwinian learning. The mechanism to do local search can be to reach a local optima or to improve the solution. Subsequently, when the individual has reached a pre-defined level, it interacts with the other members of the population through the standard EA operators. In this thesis work, Lamarckian learning is used because it has been shown to provide a faster convergence than Baldwinian learning in many works of the literature [24], [25].

Local search [26] used in MA is the process of searching in the neighborhood region of a solution and adopting a better solution if it exists. Different local methods are distinguished by their choice of search directions. In general according to algorithms that use derivative or not, continuous parametric local search methods can be categorized as zeroth, first or second order method.

(i) Zeroth order method: Also known as direct search techniques, are often very useful when the derivative information, both Hessian (the second derivative matrix) and gradient (the first derivative vector) of the function, are either unavailable or unreliable. The main feature of zeroth order method is that it only needs the object function values, or only needs the relative rank of objective values.

(ii) First order method: When the second derivative is not available, only function values and first (partial) derivative vector are used. It includes Steepest Descent, Conjugate Gradients, Quasi-Newtonian and Sequential Quadratic Programming (SQP) Methods, etc.

(iii) Second order method: This method requires the function values, its gradient and Hessian. Newton-Raphson method [27] is an example of this kind.

2.2 Surrogate-Assisted Evolutionary Algorithm

A continuing trend in science and engineering is the use of increasingly accurate simulation codes in the design and analysis process so as to produce even more reliable and

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

high quality products. Modern high-fidelity analysis codes represent some of the recent technologies that now play a central role in aiding scientists validate crucial designs and study the effect of altering key design parameters on product performance with astonishing accuracy. Nonetheless, the use of accurate simulation methods is often very timing consuming, leading to possibly unrealistic design cycle. For example, in a variety of contexts such as drug design, aerospace design, multidisciplinary structural system, rainfall prediction, establishing the quality of a potential solution can now take from many minutes to hours or months of supercomputer time. Hence, the overwhelming part of the total run time in such optimization problems is taken up by runs of the computationally expensive simulation codes.

This poses a serious impediment to the practical application of existing optimization methods for automatically establishing the critical design parameters present in real world problems in science and engineering. As a modern stochastic optimization method, EA typically requires many thousands of function calls to the simulation codes in order to locate a near optimal solution.

Using surrogate models, also referred as approximation models or meta-models, the computational burden can be greatly reduced since the efforts involved in building the surrogate models and optimization using it is much lower than the standard approach of directly coupling the simulation codes with the optimizer. The benefits of using surrogate models in engineering design optimization are elaborated as follows [11]:

(i) Improve the efficiency of optimization greatly.

(ii) Provide engineers insight to the optimization problem.

(iii) Handle both continuous and discrete variables.

(iv) Attain the parallelism easily as it involves running similar simulation at multiple design points.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

(v) Render a view of the entire design space.

In conventional numerical or local search optimization methods, it is a standard practice for computationally cheap surrogate models to be used in lieu of exact objective (fitness) functions to reduce computational cost. Several efforts on the use of surrogate-assisted conventional numerical optimization methods have been applied with much success to complex engineering optimization problems [28], [29], [30], [31]. Simpson *et al.* [32] gave a very focused review on surrogate modeling techniques and surrogate model based design optimization by going through many popular sampling methods (or experimental design methods), surrogate modeling techniques, and applications. Guidelines and recommendations were also given at the end of the paper. A summary of the panel discussion in the 9th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis & Optimization was archived in [33].

A literature study reveals that few studies have addressed the issue of incorporating surrogate models in evolutionary searches. The reason being that since EAs make use of probabilistic recombination operators, controlling the step size of design changes (to control the accuracy of approximate fitness predictions) in any SAEA will not be as straightforward as in conventional numerical optimization algorithms. Hence earlier research efforts relating to EAs optimization have focused on the use of problem specific knowledge to increase the computational efficiency [34], [35]. Robinson and Keane demonstrated the use of variable-fidelity analysis models in ES for aeronautical design [36]. It was shown that such problem specific heuristics could be effectively used to achieve performance improvements but there are finite limits to the improvements achievable by such techniques. In addition, these frameworks are restricted to a special class of surrogate models that are domain specific.

Like other population-based algorithms, EAs require a very high number of fitness function evaluations. In most real world applications the process of fitness evaluation is

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

very expensive and time consuming. The application of surrogate modeling techniques in evolutionary algorithm can decrease the number of expensive fitness evaluations required, which results in a better convergence rate of the algorithm. As a result, surrogate-assisted evolutionary optimization is receiving ever interesting attention over the recent years. There are also some surrogate-assisted hybrid methods which combine both evolutionary optimization method and conventional numerical optimization method, i.e,. both global and local search method, with surrogate models [37], [8]. This class of evolutionary optimization method that employs computationally cheap approximation models to enhance evolutionary search efficiency is often referred to as surrogate-assisted evolutionary algorithm or SAEA in short.

PseudoCode 3 Outline of a General SAEA Optimization Framework

Generate a database containing some exact design points (*Optional: upload a historical database if one exists*).
$gen := 0$; /* Initialize the generation counter. */
Create an initial population $P[gen]$.
while (*computational budget is not exhausted*) **do**
- Evaluate all individuals in the population using the exact fitness function $f(\mathbf{x})$ for a pre-defined number N of EA generations.
- Update the database with any new exact evaluated design points.
- Construct a surrogate model $\hat{f}(\mathbf{x})$ based on the exact design points in the database.
- Proceed according to the model management method employed in the SAEA by using surrogate model $\hat{f}(\mathbf{x})$.
- Apply standard EA operators to create a new population $P[gen + N + 1]$.
- $gen := gen + N + 1$; /* Increase the generation counter. */

end while

The outline of a typical SAEA optimization framework is shown in PseudoCode 3. In the first step, a database is initialized using some exact design points, which are generated either randomly or using sampling techniques. All the design points thus generated and the associated exact values of the objective and constraint functions are then archived in the database that will be used later for constructing global or local surrogate models. Alternatively, one could also use a database containing the results of a previous search

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

on the problem or a combination of the two or the database builds dynamically up as the search progresses. Subsequently, a pre-defined EA search generations are finished by using the exact fitness function $f(\mathbf{x})$, the results of any new exact fitness obtained are added into the database, facilitating later updating of surrogate models through online learning. Then a surrogate modeling technique is selected to construct a surrogate model $\hat{f}(\mathbf{x})$ with some of exact design points in the database, and the search proceeds according to the model management method employed in the SAEA by using surrogate model $\hat{f}(\mathbf{x})$. Finally, a new population is created by applying the standard EA operators. This SAEA search process is continued until the computational budget is exhausted or a user specified termination criterion is met. In this thesis, three modeling techniques are investigated, the detail mathematical formulations for RBF, PR and GP modeling techniques are provided subsequently in Chapters 3 and 4.

In general, most existing strategies employed in SAEA typically involve considering one or more of the following issues listed below:

(i) Sampling: How to sample data for computer experiments?

(ii) Model Selection: How to select an appropriate surrogate modeling technique to approximate the exact objective function, which will guide the evolution search towards better convergence?

(iii) Model Management: How to couple the surrogate model used with the evolutionary algorithm and control how the optimization process is affected by replacing the expensive real fitness evaluation by the surrogate model?

(iv) Parallelism: How to parallelize the existing SAEAs?

(v) Convergence: How to guarantee the convergence of the SAEAs?

(vi) Dimension: How to solve the high dimensional problems?

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

(vii) Evolutionary Operator: How to configure the evolutionary operators?

Among the above research issues, model selection and management are two major issues to be considered when designing SAEA for solving computationally expensive optimization problem. In what follows, a review of the existing works on surrogate modeling techniques and model management methods in SAEAs are presented.

2.3 Surrogate Modeling Techniques in SAEA

Surrogate model [4] is a (statistical) model that is built to approximate computationally expensive objective function by the simulation of analysis codes during CED optimization process. It is computationally cheap to run, and can be used in lieu of exact analysis during optimization process. In addition, it may yield insights into the functional relationship between the input $\mathbf{x}$ and the output t. If the true nature of a computer analysis code is represented as

$$t = f(\mathbf{x}) \tag{2.1}$$
$$\mathbf{x}_l \leq \mathbf{x} \leq \mathbf{x}_u$$

where $f(\mathbf{x})$ is a computationally expensive objective fitness function. $\mathbf{x}_l$ and $\mathbf{x}_u$ are lower and upper bound of input design. Then a surrogate model is an approximation of the form

$$\hat{t} = \hat{f}(\mathbf{x}) \tag{2.2}$$
$$\mathbf{x}_l \leq \mathbf{x} \leq \mathbf{x}_u$$

where $\hat{f}(\mathbf{x})$ is a computationally cheap surrogate model used to approximate the exact fitness function. $\mathbf{x}$ and $\hat{t}$ are the corresponding input and output variable, respectively.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

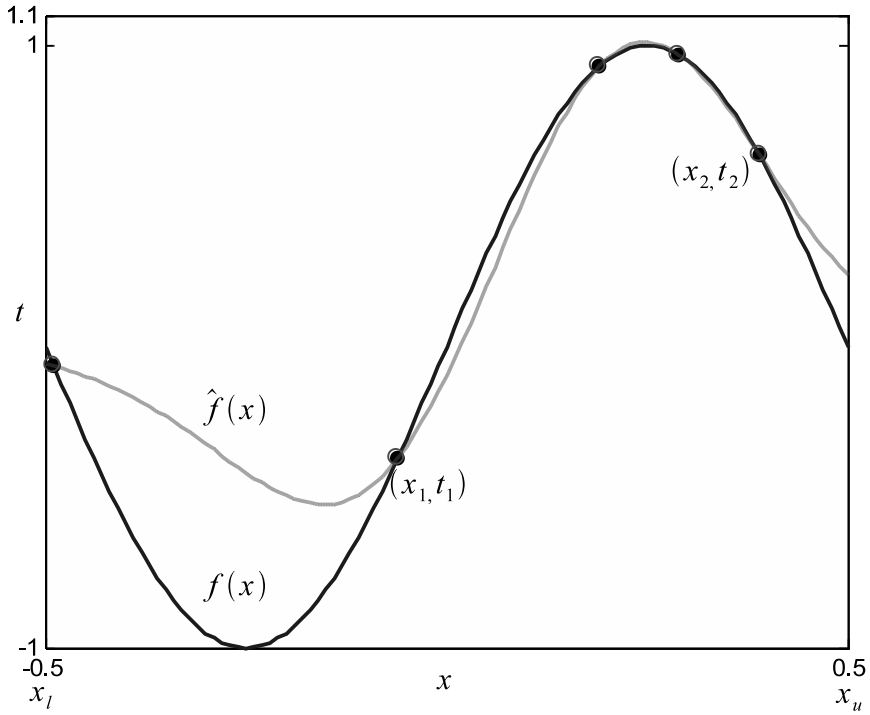


Figure 2.4: Concept of building surrogate model for computer simulations.

The concept of building surrogate model for computer simulations is shown in Figure 2.4. The region of interest is often referred to as the design space, which is bounded by the upper limit $\mathbf{x}_u$ and lower limit $\mathbf{x}_l$ of each of the design (input) variables being studied. Design of experiments strategies are often used to sample the design space to generate sample data to fit a surrogate model to each of the output variables (responses) of interest, for example, the sample data of (x_1, t_1) and (x_2, t_2).

Given n training data points with input x_i and output t_i , we are looking for an interpolation surrogate modeling technique of $\hat{f}$ so that

$$t_i = \hat{f}(x_i) \tag{2.3}$$
$$i = 1, 2, \ldots, n$$

Then the interpolation surrogate modeling technique of $\hat{f}$ can be used to approxi-

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

mate the relationship between the design (input) variables $\mathbf{x}$, and the output variable (response) t.

In recent years, there has been a growing interest in methods of applying surrogate models to engineering design optimization in which the high-fidelity analysis codes are treated as black-box functions [38]. Data collected via evaluation of the computationally expensive analysis codes at predefined design sites, often chosen by the method of design of experiment in [39]. A comprehensive review of surrogate modeling applications can be found in [4], [11], [40].

2.3.1 Surrogate Model Selection in SAEA

A variety of techniques for the constructions of surrogate model has been used in engineering design optimization [41], [42]. One popular approach in the design optimization literature is least-squares regression using low-order polynomials, also known as polynomial regression model or response surface methods [43]. A statistically sound alternative for constructing surrogate models of deterministic computer models is Kriging [44], which is referred as design and analysis of computer experiments model in the statistics literature [45] and Gaussian process in the neural networks literature [46]. Artificial neural networks, including multi-layer perceptrons and multivariate adaptive regression splines [47] have also been employed for surrogate modeling in engineering design optimization. Based on interpolation techniques, radial basis functions networks [42] are developed in the field of neural networks which are drawing more and more attention of researchers.

Several research issues of surrogate modeling in SAEA remain open. An immediate question that a designer may have is on what basis the various modeling techniques should be used. Moreover, is one surrogate modeling technique superior to the others? Numerous examples that demonstrate the application of one surrogate modeling technique or the other, typically for a specific application exist. There are several papers

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

that compare the performance of different surrogate modeling techniques. Simpson *et al.* [44] [48] compared the Kriging methods against polynomial regression models for the multidisciplinary design optimization of an aerospike nozzle. An explorative comparison was made in [49] between polynomial regression and Kriging interpolation models using a couple of benchmark test problems. A comparative study between neural network and response surface models was also provided by using a preliminary aircraft design problem in [50]. In [51], Artificial neural network methods are compared with polynomial regression models for the engine design problem in modeling the nonlinear thermodynamic behavior. In [52], polynomial regression, radial basis functions, kriging models, and multivariate adaptive regression splines are systematically compared on a variety of test problems based on multiple measures of merit (e.g., accuracy, robustness, transparency, etc.). While no one approximation model dominated, recommendations based on problem size, degree of nonlinearity, and availability of sample data are given. Further extension by Simpson [53] compared both experimental design strategies and approximation model types. Vaidyanathan *et al.* [54] presented a study comparing radial basis neural networks and response surfaces while designing the liquid rocket injector. Queipo *et al.* [55] recently reviewed different surrogate models used in the aerospace industry.

It is very useful to provide explicit rules on surrogate model selection. However, no clear conclusions can be drawn which surrogate modeling technique is found to be the most effective on all optimization problems. Among all existing surrogate modeling techniques, Polynomial Regression (PR) model, Artificial Neural Networks (ANN), Radial Basis Function (RBF) networks, and Gaussian Process (GP) model are the most prominent and commonly used modeling techniques in evolutionary optimization.

2.3.2 Global vs Local Surrogate Model

Both global and local surrogate models have been considered in the literature. Global surrogate model uses all evaluated points or the data points from entire design space. On

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

the other hand, local surrogate model only takes into account archived data points from a certain region of search space. Hence, the approximation is only valid in the vicinity of the current design point.

It is straightforward to imagine that a global model is able to simplify the search process if the surrogate model does not change the landscape of the exact fitness function. However, the local model has a number of advantages compared to global model. For complicated (e.g., multimodal, discontinuous) function landscapes it may not be feasible to build an accurate global model at all due to the effect of *'curse of dimensionality'* [42]. Further, since they use less training data, local model typically carries a far smaller computational cost.

From the view point of modeling, to build a local model is much more feasible than to build a global model. Although much work has shown that local models are generally preferred, whether a local or a global model serves its purpose most remains an open research issue.

2.4 Model Management Methods in SAEA

The use of surrogate models can be integrated into almost every stages of EA, including initialization, crossover, mutation and fitness evaluations [56]. More specifically, there has been work in EA that employs surrogate models for initializing the population [57], for reducing randomness in crossover and mutation [57], [58], [59], [60], [61], as well as using surrogate models for fitness evaluations [6], [8], [62], [63], [64], [65]. The basic motivation is to reduce the number of computationally expensive fitness evaluations without degrading the quality of the obtained optimal solution, even more, accelerating the optimization process.

Over the years, a series of SAEA frameworks using different approximation techniques have been introduced. Ratle [64] and El-Beltagy [63] examined strategies for integrating

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

evolutionary search with global surrogates based on Kriging. Various strategies using GP global surrogates have also been considered in [62], [65]. However, since the idea of constructing accurate global surrogates might be fundamentally flawed due to the '*curse of dimensionality*', online local surrogates using RBF was considered in [6], [8], [66], [67] and [68], in place of global models. In [69], neural network ensemble models was also considered for improving the quality of fitness evaluation used in the SAEA. The use of approximation models in the context of multi-objective evolutionary optimization of computationally expensive problems can also be found in [67], [70], [71] and [72]. A comprehensive review of fitness approximation in evolutionary computation is presented in [73]. A continuously updated collection of references is also available on the internet [74]. For a review of existing SAEA optimization frameworks for high-fidelity engineering design problems, the reader may refer to [75].

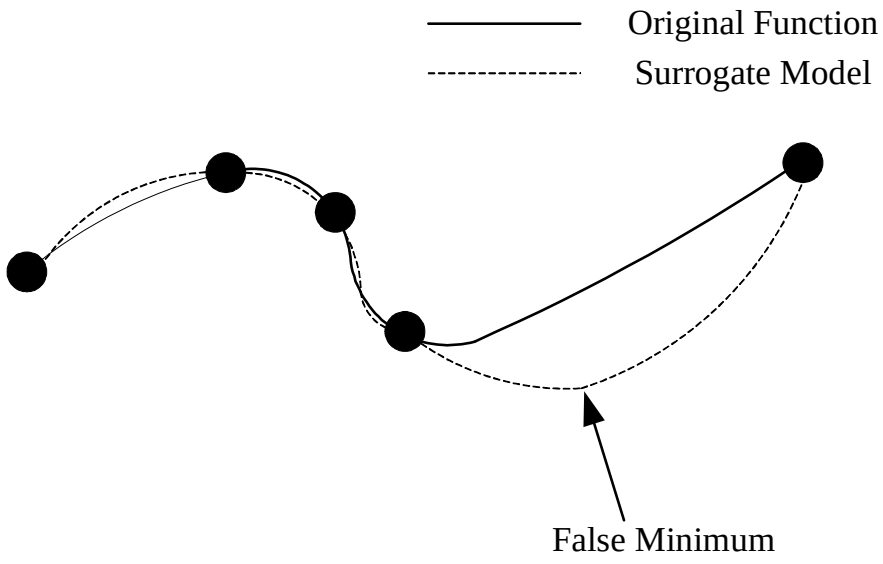


Figure 2.5: An example of a false minimum in the surrogate model (From [71]).

However, the application of surrogate models to evolutionary algorithm, i.e., SAEA, is not as straightforward as one may expect. Usually, it is very difficult to construct

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

an surrogate model that is globally correct due to the high dimensionality, complex landscape and limited number of training data points. Meanwhile, it is very likely that the evolutionary algorithm may converge to a false optimum if surrogate models are used for fitness evaluation. As Figure 2.5 shows (obtained from [73], the dots denotes the available training data points.), a false optimum is an optimum of the surrogate model, which is not one of the exact fitness function.

As a result, it is very important in most cases that the surrogate model should be used together with the exact fitness function. This can be regarded as the issue of *model management* as above mentioned. There are several *model management* methods to integrate computationally cheap surrogate models for fitness evaluation into an evolutionary algorithms. Three main types of *model management* or *evolution control* considered as below:

(i) Individual-based evolution control [76]: A fraction of individuals within a generation are evaluated on the exact fitness function itself, this part of individuals are called *controlled individuals*. If the controlled individuals are chosen randomly, it is a *random strategy*. If the top ranking individuals are chosen, it is known as a *best strategy*.

(ii) Generation-based evolution control [76]: In every L generations, $N(N < L)$ generations are controlled, i.e., all the individual are evaluated with the exact fitness function. The generations are also referred as *controlled generations*.

(iii) Memetic-based evolution control: When using surrogate models in memetic algorithm, it is also referred as *memetic-based evolution control* in this thesis. Here, surrogate models can be used to replace the exact objective functions during global/local search phase, or both search phases of the memetic algorithm.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

2.4.1 Individual-based Evolution Control

In individual-based evolution control SAEA, some of the individuals in each generation employ the surrogate model while others use the original objective function for fitness evaluations. The key problem is which of the individuals should be controlled (i.e., undergo exact evaluation).

The fraction of controlled individuals in individual-based evolution control may vary from 10% in [67] to 50% in [77]. It was suggested by Jin *et al.* [77] that when controlling individuals randomly, about 50% of the offspring need to be controlled. On the other hand, when the best strategy is used to control individuals, Jin *et al.* [77] suggest to control about 40% of the population. However, Emmerich *et al.* [67] show that for simple problems such as, a symmetric quadratic function, controlling the best 10% of pre-evaluated individuals would be sufficient. In another instance, Won *et al.* [78] only control the best individual of the entire population which is also referred as *controlled elitism*.

Furthermore, in individual-based evolution control, the individuals in the EA population are pre-evaluated by using surrogate model constructed to identify search region containing better solutions represented by the superior individuals. Hence, several pre-selection criteria for the pre-evaluation operation have been developed. Nevertheless, all approaches seems to agree that the selection criterion should represent the quality of the individual and only the most promising individuals should be selected as representatives that will be later evaluated using the original objective function.

There are various existing pre-selection criteria. An obvious and common pre-selection criterion is to use the predicted fitness value, also referred as Mean of Model Prediction (MMP). Giannakoglou investigated the pre-selection criterion of using mean prediction and proposed a new algorithm in the course of a GA-based optimization method, which is also referred as GA with Inexact Pre-Evaluation phase (GA-IPE) in [6]. This algorithm

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

utilizes dual evaluation phases in every generation. In the first Inexact Pre-Evaluation (IPE) phase, the RBF local surrogate models are build for every individual to pre-evaluate the entire population, which is a low-cost process. Then in the second phase, a predefined percentage of the current population, consisting of individuals with the top ranking mean prediction values, undergoes exact evaluations.

As a further extension, the standard RBF model used in GA-IPE are modified by taking into the account the sensitivity of the cost or fitness function with respect to each one of the design parameters, also referred as Importance Factors (IFs), giving rise to the scheme denoted as GA-IPE-IF. Furthermore, the concept of IPE is transformed to a Distributed Genetic Algorithm (DGA), a new scheme of D(GA-IPE-IF) is also proposed, known as the *best strategy*, and chooses a pre-defined number of top ranking individuals as the controlled individuals [7]. Given the superiority of GA-IPE-IF of DGA with respect to conventional GAs, D(GA-IPE-IF) appear to be more efficient and also display better exploration abilities than GA-IPE-IF and DGA.

Another pre-selection criterion is choosing the most uncertain individuals. There are two reasons to choose the most uncertain individuals. Firstly, the uncertainty introduced by surrogate models may be reduced as much as possible by re-evaluating the most uncertain individuals. Secondly, the most uncertain individuals are often located in the unexplored area. As a result, it will balance the exploration with exploitation by re-evaluating the most uncertain individuals. In [79], the distance from the concerned individual to the nearest data point for creating the surrogate model is calculated as the measure of uncertainty. In [80], the combination of the quality and uncertainty criteria has also been used to choose promising individuals.

Some surrogate modeling techniques, such as GP modeling, not only provides the mean prediction, but also gives an uncertain measure of the standard deviation. Furthermore, the Probability of Improvement (PoI) at new data point can also be derived

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

from the mean prediction and standard deviation. A data point with a larger PoI output indicates it has a higher probability of being filtered than the others. Using the mean prediction, standard deviation and PoI, a variety of pre-selection criteria for controlled individuals may be formulated. Using only the mean prediction to identify the most promising individuals comes along with one major drawback. On multi-modal problems with many misleading local minima, mean prediction tends to converge premature at those local minima. Taking this cue, Jones *et al.* [38] proposed an alternative based on the standard deviation of GP model. Beltagy and Keane [81] control the individuals whose predicted standard deviation of GP model exceeds a certain threshold, and assume the model prediction to be accurate otherwise. The threshold is decreased linearly over the course of the evolution.

In [10], a real-coded GA was coupled with Kriging for firtree structural optimization. Instead of using best strategy which choose top ranking individuals to undergo exact evaluation, a 3σ principle derived based on both the mean prediction and standard deviation is used to suggest new sample points for exact re-evaluation. The new sample points are then inserted into an ordered database storing all historical exact solutions, and surrogate model is updated when these new points fall into the part that are used in the construction of the surrogate model.

To balance exploration with exploitation of evolutionary search, Torczon and Troset [82] proposed a merit function f_M which adds a density measure to the mean prediction $\hat{f}(\mathbf{x})$. The density measure is often using the predicted standard deviation σ of GP model. Then the merit function f_M [82] is defined as:

$$f_M = \hat{f}(\mathbf{x}) - \alpha\sigma(\mathbf{x}) \tag{2.4}$$

where $\alpha \geq 0$ balances exploration with exploitation by scaling the density measure (α must be negative for maximization problem). The main idea of the merit function is to

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

increase the number of evaluations in less sampled regions of the search space by directing the search towards them. Emmerich *et al.* [67] demonstrated that merit function f_M is also a good criterion in the context of individual-based evolution control SAEA using their proposed Metamodel-Assisted Evolution Strategy (MAES). The major drawback of using merit function is to find an appropriate α, because the performance of this approach is highly dependent on α, especially for high dimensional problems.

Further, Ulmer *et al.* [65] suggest the use of PoI in comparison with using MMP. Empirical result shows that both the MMP and PoI approach enhance the performance of a standard ES on unimodal problems. Meanwhile, MMP performs slightly better than PoI, due to the higher tendency of PoI to sample in unexplored areas. On the other hand, for multimodal problems the usage of the simple MMP pre-selection fails to improve search performance. In such cases, the more sophisticated PoI pre-selection criterion appears to succeed well. It displays a higher convergence rate and is shown to be more tolerate to premature convergence than MMP. In [83], the predictions and their confidence intervals predicted by gaussian random field metamodel is used in the MAEA. Various pre-selection criterions, including MMP selection, PoI selection, expected improvement selection and lower bound selection, are compared by using MAEA. It shows that the extensive use of the uncertainty information of predictions for screening the candidate solutions makes it possible to significantly reduce the computational cost of single- and multi-objective EA.

Another interesting pre-selection criterion is to choose the most representative individuals for re-evaluation using the original fitness function according to the location of the individuals. This approach is to group the population into a number of clusters and then the individual closest to the center of the clusters selected as the representative one [69], [84].

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

2.4.2 Generation-based Evolution Control

In generation-based evolution control SAEA, the entire population is evaluated using either the surrogate model or exact fitness function. This approach is preferred if the EA is implemented in parallel. The main issue is how to determine a good control frequency of using surrogate model for fitness evaluation. There are two basic types of control methods: fixed control frequency and adaptive control frequency.

Several attempts have been made in the last few years to tackle the problem of using generation-based evolution control with fixed control frequency in SAEA. Ratle [9] examined a strategy for integrating GAs with Kriging models. In the first generation, the standard GA proceeds; subsequent generations are evaluated only on the model when the model predictions fail to improve search performance for a pre-specified number of generations, upon which the subsequent generations switch back to using the exact model. Compared to the original GA, this approach has been shown to accelerate search convergence on uni-modal and multi-modal test functions. The same problem was revisited by El-Beltagy *et al.* [63], where it was argued that the issue of balancing the concerns of optimization with that of design of experiments should be addressed. The main idea is to maintain the diversity of the individuals and to select those data points that are not redundant for model updating (online learning). In this method, the decision of when to carry out the online learning of the surrogate models is simply based on a prescribed generation delay.

Büche *et al.* [62] defined a new method which was labeled as the *surrogate approach*. In this approach, it only evaluates predicted optima on the fitness function, otherwise using the model as a surrogate. In the surrogate approach, a fitness function model is constructed for an initial training set of evaluated points. An optimization algorithm then searches for the optimum of the model's fitness prediction. The predicted optimum constitutes an ideal candidate for an improved solution to the problem, and is

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

therefore evaluated on the fitness function. The result of the evaluation is added to the model's training data, facilitating an improved approximation of the fitness function by the model. The procedure then iterates by searching for the optimum of the improved model. By using the *surrogate approach*, Büche *et al.* [62] proposed a Gaussian process optimization procedure based on 4 merit functions. These merit functions constitute different compromises between exploration and exploitation, thus balancing the inherent conflict between converging to and escaping from local minima.

There are also several attempts to use adaptive control frequency. It is straightforward to imagine that the frequency of evolution control should depend on the fidelity of the surrogate model. Jin *et al.* [85] used the exact fitness function for evaluation in the following generations until the error between model prediction and fitness function drops below a pre-defined threshold. The population then remains uncontrolled for a given number of generations before control resumes. Alternatively, Jin *et al.* estimated the local fidelity of surrogate model to adjust the frequency at which the original function is called and the surrogate model is updated by coupling ES with neural network-based surrogate models in [86]. In their approach, it was shown that the online learning of the neural networks can be improved by using the information from the covariance matrix of the evolution strategy. Numerical results are presented for two benchmark test problems as well as an aerodynamic design example which demonstrate that the proposed algorithm is able to achieve a correct solution as well as a significantly reduced computation time. Dennis *et al.* [87] suggested a method to adjust the frequency of evolution control based on the trust region framework when the generation-based approach is used. As shown by Alexandrov *et al.* [88], the trust-region strategy for adaptively controlling the move limits guarantees global convergence under some mild assumptions on the accuracy of the surrogate model.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

2.4.3 Memetic-Based Evolution Control

In memetic-based evolution control, a hybrid evolutionary algorithm, particularly a memetic algorithm is used together with surrogate models during optimization search. The surrogate models can be used in either the global or local search phase or both phases of MA.

A strategy for coupling ES with local search and quadratic response surface methods, which is referred to as landscape approximation and local search, was proposed in [37]. It does so by first mapping an approximated solution to its exact solution, followed by concentrating search efforts around the direct neighborhoods of the mapped exact solution to produce better solutions using the computationally expensive high-fidelity analysis solver. However the use of the exact analysis codes to perform local searches results in significantly high compute cost. Further by discarding the design variables interaction terms in the quadratic polynomial surrogate model may result in poor accuracy of the surrogate model, especially when working with high dimensional problems.

Ong use [8] proposed a hybrid evolutionary optimization method that employs a trust-region approach in the local search as a new scheme of Surrogate-Assisted Memetic Algorithm with Trust Region Framework (SAMA-TRF). The essential backbone of this SAMA-TRF framework is an evolutionary algorithm coupled with a feasible sequential quadratic programming solver in the spirit of Lamarckian learning. At the same time, the trust-region approach was employed for interleaving use of exact models for both the objective and constraint functions with computationally cheap RBF surrogate models during local search so as to guarantee convergence to the true local optima of the exact fitness function. Since gradient-based local search algorithm involved in EAs makes use of line searches to locate a new iterate, the issue of range of validity of the surrogate models or the control of approximation errors is directly addressed by using *ad hoc* move limits or a trust region framework.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

2.5 Summary

This chapter begins with an introduction of EAs, including GA and MA. The core concepts of SAEA are then provided. A detail and comprehensive literature survey on existing model management methods in SAEA for solving costly engineering design optimization problem is then provided.

From the literature review, two core issues in SAEA that represent some promising research areas have been found to remain relatively unexplored. These include:

(i) The use of multiple surrogate models in SAEA. From present survey, it is clear that no single surrogate model is found to be the most effective on all optimization problems. Hence, it would make good sense to consider the use of multiple and dissimilar surrogate models in SAEA.

(ii) The use of memetic-based evolution control in SAEA. Since this approach can benefit from both global and local search as well as global or local surrogate models, it is expected to lead to novel SAEA that is capable of attaining good designs at significantly lower computational budget.

Taking these cues, the use of single and multiple surrogate models in memetic algorithm is proposed and investigated in this thesis work for solving computationally expensive optimization problems under a tractable computational budget. The details are reported in the subsequent chapters.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Chapter 3

A Study on Surrogate-Assisted Memetic Algorithms

3.1 Introduction

In this chapter, a study on two variants of SAMA, namely, the standard SAMA and SAMA-TRF, is presented. In particular, the effects of the genetic operators on the SAMA search performances are investigated.

3.2 Variants of Surrogate Assisted Memetic Algorithm

In this section, two SAMA variants, i.e., the standard SAMA and SAMA-TRF, are briefly introduced. As mentioned in Chapter 1, the general nonlinear programming optimization problem is considered with the form:

$$\begin{aligned} \textbf{Minimize}: \quad & f(\mathbf{x}) \\ \textbf{Subject to}: \quad & g_i(\mathbf{x}) \leq 0, i = 1, 2, \ldots, p \\ & \mathbf{x}_l \leq \mathbf{x} \leq \mathbf{x}_u \end{aligned} \tag{3.1}$$

The focus is on cases where the evaluation of $f(\mathbf{x})$ and $g(\mathbf{x})$ are computationally expensive and it is desired to obtain a near optimal solution under a tractable computational budget.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

3.2.1 Standard SAMA

PseudoCode 4 Comparison of the standard MA and SAMA

```
Generate a database containing some exact design points.
gen := 0; /* Initialize the generation counter. */
Create an initial population P[gen].
while (computational budget is not exhausted) do
  Evaluate P[gen] using the exact fitness function f(x).
  for (each non-duplicated individual in the EA population) do
    MA: Apply local searches using exact fitness function f(x).
    SAMA: Apply local search strategy using online surrogate model f̂(x).
    Replace the individual in the population with the locally improved solution.
  end for
  Apply standard EA operators to create a new population P[gen + 1].
  gen := gen + 1; /* Increase the generation counter. */
end while
```

A unique property of MA is the heavy use of the local searches throughout the entire evolutionary search. Recent studies on MA have revealed their successes on a wide variety of real world problems. If designed correctly, they should converge to high quality solutions more efficiently than conventional evolutionary algorithms [22].

The standard MA considered here is a hybridization of the GA and a local search solver. All chromosomes in the GA population undergo local learning in the MA. Note that no form of approximations is employed in the standard MA. The core idea of the SAMA is to reduce the number of calls to the computationally expensive function by replacing the exact fitness function $f(\mathbf{x})$ used in the local search phase of the standard MA with computationally cheap surrogates $\hat{f}(\mathbf{x})$. The core differences between the standard MA and SAMA are highlighted in PseudoCode 4. Further, the basic steps of the standard SAMA optimization framework are outlined in Procedure 5.

For the sake of brevity, the standard SAMA optimization framework is presented as three main phases:

Phase 0 : In the first step, a population of evenly distributed and stratified chromosomes is initialized using design of experiments techniques such as Latin hypercube

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Procedure 5 The standard SAMA

```
/*Phase 0: Initialization */
gen := 0; /* Initialize the generation counter. */
pop[gen] := CreateInitPop(db); /*Create an initial population from database*/
while (computational budget is not exhausted) do
  fitness[gen] := Evaluate(pop[gen], f(x)); /*Evaluate population using f(x)*/

  /*Phase 1: Local Search Strategy */
  for (each non-duplicated individual in pop[gen]) do
    f̂(x) := BuildLocalModel(db); /*Construct a online local surrogate model*/
    LIpop[gen] := LocalImprovement(f̂(x)); /*Apply local searches with f̂(x)*/
    LIfitness[gen] := Evaluate(LIpop[gen], f(x)); /*Evaluate local improvements*/
    /*Replace the individual with locally improved solution.*/
    pop[gen] ⇐ LIpop[gen];
    fitness[gen] ⇐ LIfitness[gen];
  end for

  /*Phase 2: Standard EA Operations */
  for i = 0 to MaxPopSize do
    (parent1[gen], parent2[gen]) := Select(pop[gen]);
    child[gen] := Crossover(parent1[gen], parent2[gen]);
    Mutation(child[gen]);
  end for
  pop[gen + 1] := CreateNewPop(child[gen]).
  gen := gen + 1; /* Increase the generation counter. */
end while
```

sampling [89]. These points are evaluated using the exact fitness function $f(\mathbf{x})$. The exact fitness values obtained are then archived in a central database, together with the design vectors.

Phase 1 : The local search strategy in SAMA embeds a local search optimizer to exploit better solutions. M training data points are extracted from the archived database of design points evaluated so far using the exact computationally expensive fitness functions. These points are then used to construct online surrogates of the exact fitness functions. The surrogate models $\hat{f}(\mathbf{x})$ thus created are used to facilitate the necessary fitness function estimations in the local searches. If an improved solution is found by the

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

local search strategy, the genotype are forced to reflect the result of improvement with Lamarckian learning by placing the locally improved individual back into the population to compete for reproductive opportunities.

Phase 2 : The population then proceeds with the evolutionary operators of crossover and mutation, etc. This search process is repeated until the computational budget is exhausted, the global optimum is found or a user specified termination criterion is met.

3.2.2 SAMA with Trust Region Framework

The SAMA with Trust Region Framework (SAMA-TRF) incorporates a trust-region framework in the standard SAMA for interleaving use of exact objective function $f(\mathbf{x})$ with computationally cheap local surrogate models $\hat{f}(\mathbf{x})$ during local searches [8]. Since gradient-based local search algorithm involved in EAs makes use of line searches to locate a new iterate, the issue of range of validity of the surrogate models or the control of approximation errors can be directly addressed by using *ad hoc* move limits or a trust region framework. As shown in [88], the trust-region strategy guarantees global convergence under some mild assumptions on the accuracy of the surrogate model. The work flow of the SAMA-TRF optimization algorithm is depicted in Figure 3.1.

3.3 Local Search Strategy

As a key component in SAMAs, the local search strategy is designed to work with a locally trained system that adjusts to the local properties of the training data in each area of the input space. In the standard SAMA, the local search strategy uses local surrogate models together with the local search solver. In the SAMA-TRF, the TRF is also used in the local search strategy to switch between the expensive and approximate models during local search.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

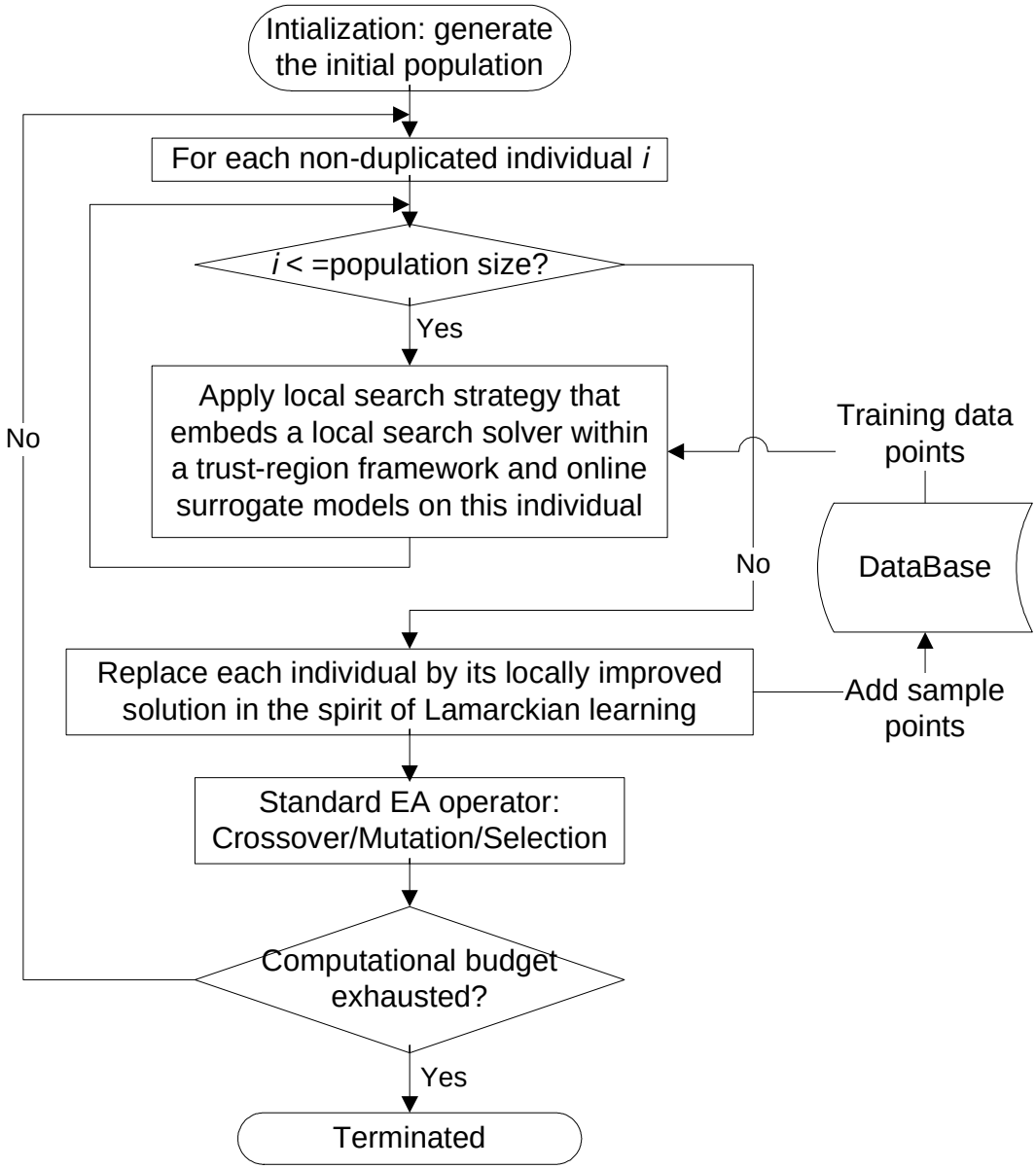


Figure 3.1: Work flow of the SAMA-TRF

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

3.3.1 Radial Basis Function

Since local surrogate models will probably be built thousands of times during the local search, computational efficiency is a major concern. This consideration motivates the use of RBF local surrogate model, which can be efficiently applied to approximate multiple-input multiple-output data, particularly when a few hundred data points are used for training. A further motivation is that RBF networks have good accuracy and fast training in [90].

Let $\mathcal{D} = \{\mathbf{x}_i, t_i\}, i = 1 \dots n$ denote the training dataset, where $\mathbf{x}_i \in \mathbb{R}^d$ is an input design vector and $t_i \in \mathbb{R}$ is the corresponding target value. Then the local surrogate models are interpolating radial basis function networks of the form

$$\hat{t} = \hat{f}(\mathbf{x}) = \sum_{i=1}^{n} \alpha_i K(||\mathbf{x} - \mathbf{x}_i||) \tag{3.2}$$

where $K(||\mathbf{x}-\mathbf{x}_i||) : \mathbb{R}^d \rightarrow \mathbb{R}$ is a RBF and $\boldsymbol{\alpha} = \{\alpha_1, \alpha_2, \dots, \alpha_n\} \in \mathbb{R}^n$ denotes the vector of weights.

Typical choices for the kernel include linear splines, cubic splines, multiquadrics, thin-plate splines, and Gaussian functions [42]. The structure of some commonly used radial basis kernels and their parameterization are shown in Table 3.1. Given a suitable kernel, the weight vector can be computed by solving the linear algebraic system of equations $\mathbf{K}\boldsymbol{\alpha} = \mathbf{t}$, where $\mathbf{t} = \{t_1, t_2, \dots, t_n\} \in \mathbb{R}^n$ denotes the vector of outputs and $\mathbf{K} \in \mathbb{R}^{n \times n}$ denotes the Gram matrix formed using the training inputs (i.e., the ijth element of $\mathbf{K}$ is computed as $K(||\mathbf{x}_i - \mathbf{x}_j||)$).

Micchelli [91] proved that non-singularity of the Gram matrix $\mathbf{K}$ can be theoretically guaranteed for a class of kernels only when the set of input vectors in the training dataset are distinct. In many papers about the radial basis function literature, a polynomial term P is often appended to (3.2) along with some constraints. In other words, if K is

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Table 3.1: Radial Basis Kernels

Linear Splines	$\|\|\mathbf{x}-\mathbf{c}_i\|\|$
Thin Plate Splines	$\|\|\mathbf{x}-\mathbf{c}_i\|\|^k ln\|\|\mathbf{x}-\mathbf{c}_i\|\|$
Cubic Splines	$\|\|\mathbf{x}-\mathbf{c}_i\|\|^3$
Gaussian	$e^{-\frac{\|\|\mathbf{x}-\mathbf{c}_i\|\|^2}{\beta_i}}$
Multiquadrics	$\sqrt{1+\frac{\|\|\mathbf{x}-\mathbf{c}_i\|\|^2}{\beta_i}}$
Inverse Multiquadrics	$(1+\frac{\|\|\mathbf{x}-\mathbf{c}_i\|\|^2}{\beta_i})^{-\frac{1}{2}}$

a conditionally positive definite function of order q, then to ensure a unique solution for the weight vector, (3.2) is rewritten as

$$\hat{t} = \sum_{i=1}^{n} \alpha_i K(\mathbf{x}, \mathbf{x}_i) + P_q(\mathbf{x}), \tag{3.3}$$

where P_q is a polynomial of order $q-1$ and the following homogeneous constraint equations are imposed

$$\sum_{i=1}^{n} \alpha_i P_j(\mathbf{x}_i) = 0, \quad 1 \leq j \leq q \tag{3.4}$$

Then the weight vector can be computed by solving a linear algebraic system of equations of the form $\mathbf{Ax} = \mathbf{b}$, where

$$\mathbf{A} = \begin{bmatrix} \mathbf{K} & \mathbf{P} \\ \mathbf{P}^T & \mathbf{0} \end{bmatrix}, \ \mathbf{x} = \{\boldsymbol{\alpha}, \boldsymbol{\theta}\}^T, \text{ and } \mathbf{b} = \{\mathbf{t}, \mathbf{0}\}^T, \tag{3.5}$$

where $\mathbf{P}$ is a matrix which arises by substituting the input vectors in the training dataset into the polynomial term P, and $\boldsymbol{\theta}$ is a vector composed of the undetermined coefficients of P. In practice, good approximations can be obtained by using a constant instead of a full-order polynomial. Here, the coefficient matrix $\mathbf{A}$ becomes

$$\mathbf{A} = \begin{bmatrix} \mathbf{K} & \mathbf{1} \\ \mathbf{1}^T & 0 \end{bmatrix} \in \mathbb{R}^{(n+1)\times(n+1)}, \tag{3.6}$$

where $\mathbf{1} \in \mathbb{R}^n$ is a vector of ones.

For problems with multiple outputs, for example, problems with multiple objectives and constraints, the weight vector can be efficiently computed for all the outputs of

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

interest once the matrix $\mathbf{K}$ is decomposed, providing indisputable computational cost savings.

The use of linear splines is considered for constructing surrogate models since earlier study in [8] suggests that this kernel is capable of providing models with good generalization capability at a low computational cost.

3.3.2 Trust Region Framework

The TRF is used in SAMA-TRF to ensure the convergence to the local optimum of the exact computationally expensive fitness function [8], [92]. More specifically, for each non-duplicated individuals in the population, the local search proceeds with a sequence of trust-region subproblems of the form

$$\begin{aligned} \textbf{Minimize}: \quad & \hat{f}^k(\mathbf{x}+\mathbf{x}_c^k) \\ \textbf{Subject to}: \quad & \hat{g}_i^k(\mathbf{x}+\mathbf{x}_c^k) \leq 0, i=1,2,\ldots,p \\ & ||\mathbf{x}|| \leq \Omega^k \end{aligned} \tag{3.7}$$

where $k = 0, 1, 2, \ldots, k_{max}$, $\hat{f}(x)$ and $\hat{g}(x)$ are the approximation functions corresponding to the objective function $f(x)$ and constraint function $g(x)$, $\mathbf{x}_c^k$ and Ω^k are the initial guess and the trust-region radius used for local search at iteration k, respectively.

For each subproblem (or during each trust-region iteration), surrogate models of the exact fitness function, viz., $\hat{f}^k(\mathbf{x})$ is created dynamically. The M nearest neighbors of the initial guess, $\mathbf{x}_c^k$, are extracted from the archived database of design points evaluated so far using the exact analysis codes. These points are then used to construct local surrogate models of the exact objective function.

The surrogate models thus created are used to facilitate the necessary fitness function estimations in the local searches. During local search, the trust-region Ω is initialized by using the minimum and maximum values of the design points used to construct the

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

surrogate model. After each iteration, the trust-region radius Ω^k is updated based on a measure which indicates the accuracy of the surrogate model at the kth local optimum, $\mathbf{x}_{lo}^k$. After computing the exact values of the fitness function at this point, the figure of merit, ρ^k, is calculated as

$$\rho^k = \min(\rho_f^k, \rho_{g_i}^k), \text{ for } i = 1, 2, \ldots, p \tag{3.8}$$

where

$$\rho_f^k = \frac{f(\mathbf{x}_c^k) - f(\mathbf{x}_{lo}^k)}{\hat{f}(\mathbf{x}_c^k) - \hat{f}(\mathbf{x}_{lo}^k)} \text{ and } \rho_{g_i}^k = \frac{g_i(\mathbf{x}_c^k) - g_i(\mathbf{x}_{lo}^k)}{\hat{g}_i(\mathbf{x}_c^k) - \hat{g}_i(\mathbf{x}_{lo}^k)} \tag{3.9}$$

The above equation provides a measure of the actual versus predicted change in the exact fitness function values at the kth local optimum. The value of ρ^k is then used to update the trust-region radius as follows [88]:

$$\begin{aligned} \Omega^{k+1} &= 0.25\Omega^k, \quad \text{if } \rho^k \leq 0.25, \\ &= \Omega^k, \quad \text{if } 0.25 < \rho^k \leq 0.75, \\ &= \xi\Omega^k, \quad \text{if } \rho^k \geq 0.75, \end{aligned} \tag{3.10}$$

where $\xi = 2$, if $||\mathbf{x}_{lo}^k - \mathbf{x}_c^k||_\infty = \Omega^k$ or $\xi = 1$, if $||\mathbf{x}_{lo}^k - \mathbf{x}_c^k||_\infty < \Omega^k$.

The trust-region radius, Ω^k, is reduced if the accuracy of the surrogate, measured by ρ^k, is low. Ω^k is doubled if the surrogate is found to be accurate and the kth local optimum, $\mathbf{x}_{lo}^k$, lies on the trust-region bounds. Otherwise the trust-region radius remains unchanged.

The exact solutions of the fitness functions at the kth local optimum are combined with the existing neighboring data points to generate new surrogate models in the subsequent trust-region iterations. The initial guess for the k+1 iteration is defined by

$$\begin{aligned} \mathbf{x}_c^{k+1} &= \mathbf{x}_{lo}^k, \quad \text{if } \rho^k > 0 \\ &= \mathbf{x}_c^k, \quad \text{if } \rho^k \leq 0. \end{aligned} \tag{3.11}$$

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

The trust-region process for an individual terminates when the maximum number of trust-region iterations permissible, k_{max}, which is defined by the user, is reached. Lamarckian learning then proceeds if the k_{max} local optimum solution obtained is an improvement over that of the initial individual.

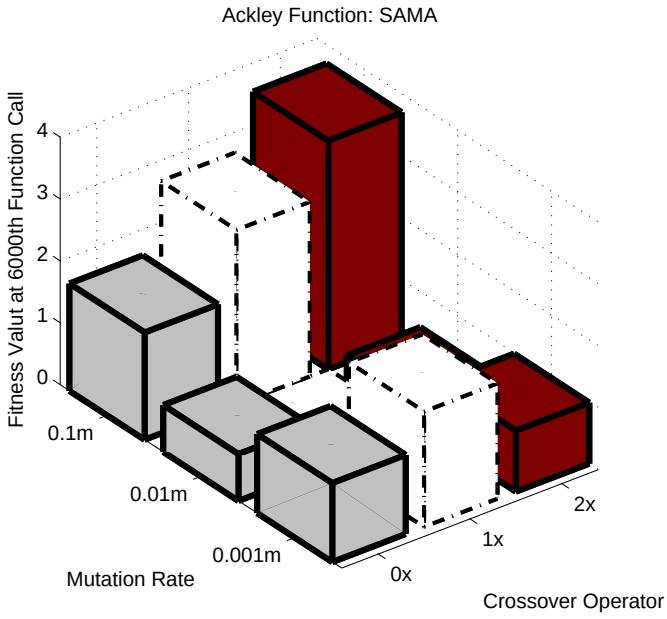
3.4 Empirical Study on SAMA Variants

Next, an empirical study on standard SAMA and SAMA-TRF for different settings of the genetic operators is presented. Several multimodal benchmark test functions, i.e., Ackley, Griewank and Rastrigin function are used in the empirical study. All benchmark test functions used in the study are of 10 dimensions and have a single global minimum at zero. The reader is referred to Appendix A for greater details of the benchmark test functions used.

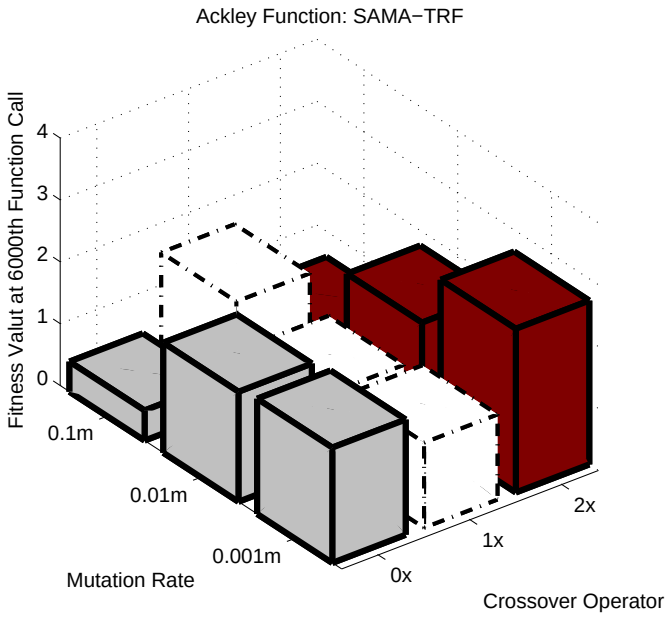
In present experimental study, a standard GA with population size of 50, crossover operator at probabilities 0.6 is employed. A linear ranking algorithm [93] is used for selection. The effects of the genetic operators on SAMAs search performances, particularly different mutation rates of 0.1, 0.01 or 0.001, and crossover procedures of 1-point/2-point/uniform crossover are studied.

Apart from the standard GA settings, the two user-specified parameters of the SAMA-TRF are - 1) number of training data points used to construct the surrogate model M, and 2) maximum trust region iterations k_{max}. In present numerical studies, set M and k_{max} to 100 and 3 respectively as suggested in [8]. The SAMA framework employs similar configurations as the SAMA-TRF. It should be noted that the M nearest neighbors of the interested individual extracted from the archived database are used to construct the surrogate model here. The criterion used to determine the similarity between design points is the simple Euclidean distance metric. Further, the Feasible Sequential Quadratic Programming (FSQP) [94] is used here as the local search solver in the SAMA variants.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library



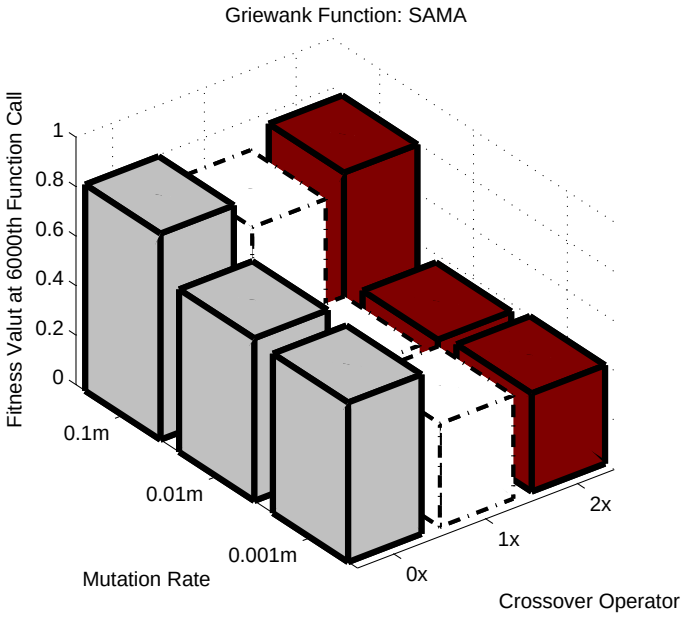
3.2.a: The effect of genetic operators on SAMA.



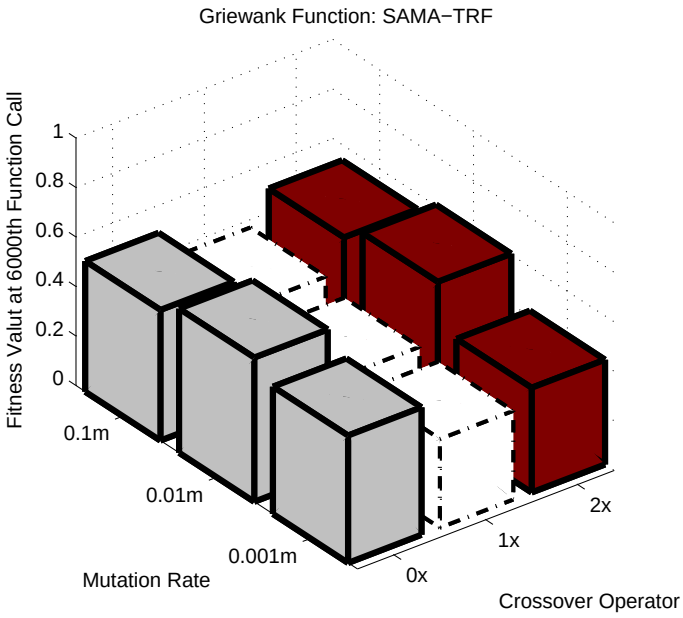
3.2.b: The effect of genetic operators on SAMA-TRF.

Figure 3.2: Average fitness value among 20 independent runs for SAMA and SAMA-TRF with different mutation rate and crossover procedure on Ackley function.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library



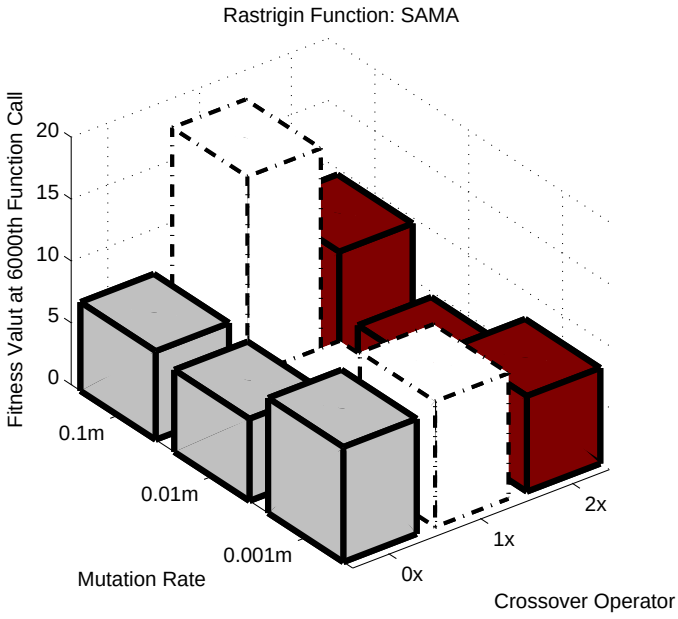
3.3.a: The effect of genetic operators on SAMA.



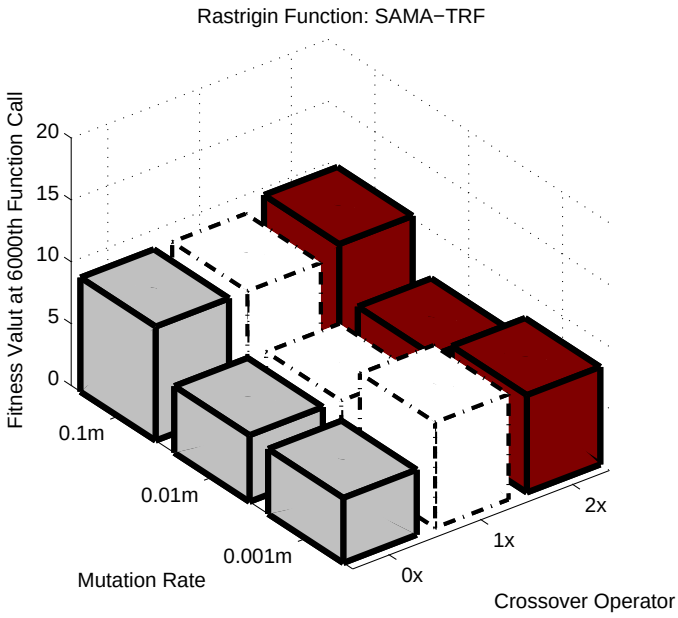
3.3.b: The effect of genetic operators on SAMA-TRF.

Figure 3.3: Average fitness value among 20 independent runs for SAMA and SAMA-TRF with different mutation rate and crossover procedure on Griewank function.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library



3.4.a: The effect of genetic operators on SAMA.



3.4.b: The effect of genetic operators on SAMA-TRF.

Figure 3.4: Average fitness value among 20 independent runs for SAMA and SAMA-TRF with different mutation rate and crossover procedure on Rastrigin function.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Mutation Rate	Ackley		Griewank		Rastrigin	
	SAMA	SAMA-TRF	SAMA	SAMA-TRF	SAMA	SAMA-TRF
0.1	2.7502	**0.9172**	0.7812	0.4902	11.6951	9.5495
0.01	**0.4976**	1.5742	**0.4654**	0.5187	**5.0046**	**5.6015**
0.001	1.3738	1.9614	0.4859	**0.4319**	9.0501	7.2084

Table 3.2: Average fitness value among 20 independent runs for SAMA and SAMA-TRF with different mutation rates on all benchmark test functions.

Mutation Rate	Ackley		Griewank		Rastrigin	
	SAMA	SAMA-TRF	SAMA	SAMA-TRF	SAMA	SAMA-TRF
0.1	1.7330	**0.4862**	0.7161	0.4067	7.1623	9.2127
0.01	**0.3131**	1.1885	**0.3671**	0.3697	**3.8399**	5.3066
0.001	0.9832	1.3762	0.3969	**0.3586**	7.7210	**5.2515**

Table 3.3: Best fitness value among 20 independent runs for SAMA and SAMA-TRF with different mutation rates on all benchmark test functions.

For a better view on the effects of the genetic operators on both standard SAMA and SAMA-TRF, the 3-D barplots of the fitness value at $6,000$ maximum function calls are shown in Figs 3.2-3.4 for different mutation rates and crossover procedures. It should be noted that all results presented are average of 20 simulation runs conducted with a tractable computational budget of (6×10^3) exact fitness function evaluations.

The results obtained in Figs 3.2-3.4 indicate that the effects of the genetic operators (both mutation rate and crossover procedure, 0x denotes uniform crossover in the figures) on the standard SAMA are more significant than SAMA-TRF (i.e., the outcome of using SAMA has much bigger difference than using SAMA-TRF). In particular, the standard SAMA appears to perform poorly for a high mutation rate of 0.1 on all three multimodal benchmark test functions. Further, there is significant impact on the standard SAMA for the 2-point crossover procedure on Ackley benchmark test function in Figure 3.2 and 1-point crossover procedure on Rastrigin benchmark test function in Figure 3.4. On the other hand, the SAMA-TRF is observed to be more robust against the changing mutation rate or crossover procedure on all the benchmark test problems considered.

In order to identify a suitable mutation rate for both SAMA and SAMA-TRF, the

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Mutation Rate	Ackley		Griewank		Rastrigin		Total Ranking	
	SAMA	TRF	SAMA	TRF	SAMA	TRF	SAMA	TRF
0.1	3	1	3	2	3	3	9	6
0.01	1	2	1	3	1	1	3	6
0.001	2	3	2	1	2	2	6	6

Table 3.4: Ranking of average fitness value among 20 independent runs for SAMA and SAMA-TRF with different mutation rates on all benchmark test functions.

Mutation Rate	Ackley		Griewank		Rastrigin		Total Ranking	
	SAMA	TRF	SAMA	TRF	SAMA	TRF	SAMA	TRF
0.1	3	1	3	3	2	3	8	7
0.01	1	2	1	2	1	2	3	6
0.001	2	3	2	1	3	1	7	5

Table 3.5: Ranking of best fitness value among 20 independent runs for SAMA and SAMA-TRF with different mutation rates on all benchmark test functions.

average values and best fitness values of standard SAMA and SAMA-TRF for different mutation rates on all three multimodal benchmark test functions are summarized in Tables 3.2 and 3.3, respectively. The value in each column is the average fitness value of using SAMA or SAMA-TRF for different crossover procedure settings (at the end of 6000 exact function calls). The best fitness value in each column is presented in bold. The corresponding ranking of the average and best fitness values for SAMA and SAMA-TRF are also depicted in Tables 3.4 and 3.5, respectively. In each column of the Tables, the best fitness value is ranked 1st.

As reported in the literature, the mutation rate has a significant effect on the search performance of the standard GA. A low mutation rate of 0.001 is often preferable in the standard GA, while a high mutation rate of 0.1 may result in poor search performance due to the increase exploration or randomness in the population. In contrast, from the ranking in Tables 3.4 and 3.5, it can be observed that a mutation rate of 0.01 attains the best solution quality for SAMA optimization framework on all test functions considered, while a mutation rate of 0.001 appears to be suitable for SAMA-TRF optimization framework.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Crossover Method	Ackley		Griewank		Rastrigin	
	SAMA	SAMA-TRF	SAMA	SAMA-TRF	SAMA	SAMA-TRF
uniform	**1.2547**	**1.3755**	0.7112	0.5414	7.6942	**6.6136**
1-point	1.6664	1.4149	**0.5017**	**0.3783**	10.7991	7.6874
2-point	1.7006	1.6624	0.5195	0.5211	**7.2565**	8.0584

Table 3.6: Average fitness value among 20 independent runs for SAMA and SAMA-TRF with different crossover procedure on all benchmark test functions

Crossover Method	Ackley		Griewank		Rastrigin	
	SAMA	SAMA-TRF	SAMA	SAMA-TRF	SAMA	SAMA-TRF
uniform	0.7565	**0.4862**	0.6440	0.5143	6.6304	**5.2515**
1-point	**0.3131**	1.1885	0.3723	**0.3586**	**3.8399**	5.3066
2-point	0.4233	0.5853	**0.3671**	0.4228	4.5434	6.1213

Table 3.7: Best fitness value among 20 independent runs for SAMA and SAMA-TRF with different crossover procedure on all benchmark test functions.

Crossover Method	Ackley		Griewank		Rastrigin		Total Ranking	
	SAMA	TRF	SAMA	TRF	SAMA	TRF	SAMA	TRF
uniform	1	1	3	3	2	1	6	5
1-point	2	2	1	1	3	2	6	5
2-point	3	3	2	2	1	3	6	8

Table 3.8: Ranking of average fitness value among 20 independent runs for SAMA and SAMA-TRF with different crossover procedures on all benchmark test functions.

Crossover Method	Ackley		Griewank		Rastrigin		Total Ranking	
	SAMA	TRF	SAMA	TRF	SAMA	TRF	SAMA	TRF
uniform	3	1	3	3	3	1	9	5
1-point	1	3	2	1	1	2	4	6
2-point	2	2	1	2	2	3	5	7

Table 3.9: Ranking of best fitness value among 20 independent runs for SAMA and SAMA-TRF with different crossover procedures on all benchmark test functions.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Consider next a suitable crossover procedure for both SAMA and SAMA-TRF. The average values (with different mutation rate setting) and best fitness values of standard SAMA and SAMA-TRF for different crossover procedures are tabulated in Tables 3.6-3.7 on all the three multimodal benchmark test functions. The ranking of the average and best fitness values are also reported in Table 3.8-3.9, respectively. It is observed the 1-point crossover appears to be more appropriate in the standard SAMA, while uniform crossover in SAMA-TRF leads to better solution quality.

3.5 Summary

In this chapter, a study on two existing SAMA variants, i.e., the standard SAMA and SAMA-TRF, is presented. In particular, the search performances of the two SAMA variants are presented and analyzed for various configurations of the genetic operators. Empirical result shows that the SAMA-TRF serves to be more robust to the effects of changes in the genetic operators than the standard SAMA. Further, a mutation rate of 0.01 and the 1-point crossover procedure is shown to generate good solution quality on the standard SAMA. On the other hand, a mutation rate of 0.001 and uniform crossover procedure appears to be most suitable for SAMA-TRF. Since it should be noted that the research focus of this dissertation work is on the use of surrogate modeling techniques in the context of evolutionary computation, the adaptation of evolutionary operators for generating new solutions are not experimented much, greater study on the effect of the genetic operators is rather considered as a possible future research direction.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Chapter 4

Hierarchical Surrogate-Assisted Memetic Algorithm

In this chapter, a novel HSAMA optimization framework combining both global and local surrogate models for solving computationally expensive optimization problems under a tractable computational budget is investigated and presented. This hierarchical optimization framework is motivated by the lack of suitable multi-layer SAMA for solving computationally expensive optimization problems. In other words, how multiple surrogate models can be synergistically combined in both global and local search phases of MA is shown to accelerate optimization search.

In contrast to existing SAEAs in the literature, the proposed HSAMA optimization framework employs a global surrogate model to identify the promising individuals in the EA population at the global search phase of MA. Subsequently, these promising individuals then undergo Lamarckian learning based on a trust-region enabled local search strategy that employs computationally cheap local surrogate models.

4.1 Outline of HSAMA Optimization Framework

The proposed HSAMA optimization framework is presented as four main phases:

Phase 0 {*Initialization*}: At the first step, some exact design points are initialized either randomly or using design of experiments techniques. These design points are

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

PseudoCode 6 HSAMA

Generate a database containing some exact design points.
$gen := 0$; /* Initialize the generation counter. */
Create an initial population $P[gen]$.
while (*computational budget is not exhausted*) **do**
  Construct a global surrogate model $\hat{f}_g(\mathbf{x})$.
  Evaluate $P[gen]$ using the global surrogate model $\hat{f}_g(\mathbf{x})$.
  Pre-select η percent promising individuals from $P[gen]$ based on $\hat{f}_g(\mathbf{x})$.
  for (each pre-selected η percent promising individual) **do**
    • Evaluate this individual using the exact fitness function $f(\mathbf{x})$.
    • Construct a online local surrogate model $\hat{f}_l(\mathbf{x})$ for this individual.
    • Apply trust-region enabled local search strategy to the individual which interleaves $f(\mathbf{x})$ with $\hat{f}_l(\mathbf{x})$.
    • Update the database with any new exact evaluated design points.
    • Replace the individual in the population with the locally improved solution.
  end for
  Apply standard EA operators to create a new population $P[gen + 1]$.
  $gen := gen + 1$; /* Increase the generation counter. */
end while

evaluated using the exact fitness function and archived in a central database together with exact fitness values obtained. Then a global surrogate model $\hat{f}_g(\mathbf{x})$ is constructed to represent the global trends of the entire fitness landscape.

Phase 1 {*Global Search Strategy*}: Subsequently, the global surrogate model $\hat{f}_g(\mathbf{x})$ is used to pre-evaluate all individuals of the population. The predictions produced by using $\hat{f}_g(\mathbf{x})$ are used to pre-screen subsequent EA populations such that only the $\eta\%$ $(0 < \eta < 100)$ promising individuals undergo Lamarckian learning. This prevents any unnecessary local searches from being conducted on individuals whose actual fitness is anticipated to be poor.

Phase 2 {*Local Search Strategy*}: A Lamarckian evolution process involving a trust-region framework is devised for the local searches. For each $\eta\%$ individual, a local surrogate model $\hat{f}_l(\mathbf{x})$ is built dynamically to represent the local fitness landscape in the vicinity of an individual. Then the Lamarckian learning process is conducted to find an improved solution. Subsequently, results of any new exact fitness obtained are added into

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Procedure 7 HSAMA

/*Phase 0: Initialization */
$gen := 0$; /* Initialize the generation counter. */
$pop[gen] := CreateInitPop(db)$; /*Create an initial population from database*/
while (*computational budget is not exhausted*) **do**

/*Phase 1: Global Search Strategy */
$\hat{f}_g(\mathbf{x}) := BuildGlobalModel(db)$; /*Construct a global surrogate model*/
$fitness[gen] := Evaluate(pop[gen], \hat{f}_g(\mathbf{x}))$; /*Evaluate population using $\hat{f}_g(\mathbf{x})$*/
/*Pre-select promising individuals of $pop'[gen] = \eta\% \times pop[gen]$ */
$pop'[gen] := PreSelectCriterion(\eta, pop[gen], fitness[gen])$;

/*Phase 2: Local Search Strategy */
for (each pre-selected $\eta\%$ promising individual in $pop'[gen]$) **do**
$\hat{f}_l(\mathbf{x}) := BuildLocalModel(db)$; /*Construct a online local surrogate model*/
$LIpop[gen] := TRFLocalImprovement(f(\mathbf{x}), \hat{f}_l(\mathbf{x}))$;/*Apply local searches*/
$LIfitness[gen] := Evaluate(LIpop[gen], f(\mathbf{x}))$; /*Evaluate local improvements*/
/*Replace the individual with locally improved solution.*/
$pop[gen] \Leftarrow LIpop[gen]$;
$fitness[gen] \Leftarrow LIfitness[gen]$;
end for

/*Phase 3: Standard EA Operations */
for $i = 0$ to $MaxPopSize$ **do**
$(parent1[gen], parent2[gen]) := Select(pop[gen])$;
$child[gen] := Crossover(parent1[gen], parent2[gen])$;
$Mutation(child[gen])$;
end for
$pop[gen + 1] := CreateNewPop(child[gen])$.
$gen := gen + 1$; /* Increase the generation counter. */
end while

the central database, facilitating possible updating of surrogate models through online learning. Here, the local search strategy is designed with locally trained system that adjusts to the local properties of the input space. Further, the use of the TRF maintains convergence close to the local optima of the original problem during local searches by interleaving the use of computationally expensive $f(x)$ with computationally cheap $\hat{f}_l(\mathbf{x})$.

Phase 3 {*Standard EA Operations*}: The population then proceeds with the standard

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

EA operators of crossover, mutation, etc. This process of HSAMA search is continued until the computational budget is exhausted or a user specified termination criterion is met.

The basic steps of the proposed HSAMA optimization framework combining both global and local surrogate models for solving computationally expensive problems are outlined in PseudoCode 6 and Procedure 7. Next, the phase of global search strategy is described in greater detail.

4.2 Global Search Strategy

The global search strategy is designed to identify search regions that contain better-quality solutions, here represented by the superior individuals among the EA population. Usually, a global surrogate model is used to pre-evaluate the entire population of individuals based on the approximated fitness value [6], [62], [67], [65]. The global surrogate model is constructed using the top ranking q archived design points of the database in the present study. Since the training data points spread across the entire search space to represent the whole fitness landscape, hence the model is termed a global surrogate model.

The intuitive choice of global surrogate model in the present framework should be one that is capable of modeling complex global trends of the exact fitness landscape accurately. A statistically rigorous approximation is the idea of Bayesian interpolation or regression. It is also referred to as Gaussian process in the neural networks literature or Kriging in the geostatistics literature and generally recognized as a powerful tool for modeling complex landscape accurately [95]. Since GP model retains the aforementioned features, it makes good sense to use it as a global surrogate model. A further motivation is that GP method is statistically more meaningful and also allows the possibility of computing error estimates for the predicted outputs. This provides a further information

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

of the accuracy of the prediction at new points and is used in our proposed hierarchical optimization procedure to find the promising EA individuals. The GP global surrogate model based HSAMA is also referred as GP-HSAMA.

On the other hand, since the purpose of the global surrogate model is to pre-screen the EA population, the accuracy of the predicted fitness may not be very critical to the HSAMA algorithm. Rather, the correct selection of promising individuals would be of upmost importance from the evolutionary search point of view. Such an observation was also reported in [96] where it was shown that the qualitative fitness approximation of the surrogate model may be sufficient in SAEA frameworks, even though the approximation error may be quite large. Taking these cues, using the PR method for building global surrogate model is also considered in the HSAMA algorithm since the PR method is computationally more efficient compared to GP model. The trade-off is of course a lower accuracy in the global surrogate obtained by using PR. The PR global surrogate model based HSAMA is also referred as PR-HSAMA.

Next, the proposed Data Parallel Gaussian Process (DPGP) and PR modeling technique are described, respectively. In addition, the pre-selection criterion used to pre-screen the population of promising individuals in the global search strategy is also introduced.

4.2.1 Data Parallel Gaussian Process Modeling

Here, the standard GP modeling technique is introduced, then the proposed DPGP modeling technique is presented.

Let $\mathcal{D} = \{\mathbf{x}_i, t_i\}, i = 1 \dots n$ denote the training dataset, where $\mathbf{x}_i \in \mathbb{R}^d$ is an input design vector and $t_i \in \mathbb{R}$ is the corresponding target value. The GP surrogate model assumes the presence of an unknown true modeling function $f(\mathbf{x})$ and an additive noise term v to account for anomalies in the observed data. Thus:

$$t = f(\mathbf{x}) + v \tag{4.1}$$

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

where $\mathbf{x}$ and t are the input and output variable, respectively.

The standard analysis requires the specification of prior probabilities on the modeling function and the noise model. From a stochastic process viewpoint, the output collection $\mathbf{t} = \{t_1, t_2, ..., t_n\}$ is called a Gaussian process if every subset of $\mathbf{t}$ has a joint Gaussian distribution. More specifically,

$$P(\mathbf{t}|\mathbf{C}, \{x_n\}) = \frac{1}{Z} \exp\left(-\frac{1}{2}(\mathbf{t} - \boldsymbol{\mu})^T \mathbf{C}^{-1}(\mathbf{t} - \boldsymbol{\mu})\right) \tag{4.2}$$

where $\mathbf{C}$ is a covariance matrix parameterized in terms of hyperparameters $\boldsymbol{\theta}$, i.e., $\mathbf{C}_{ij} = k(x_i, x_j; \theta)$ and $\boldsymbol{\mu}$ is the process mean. The Gaussian process is characterized by this covariance structure since it incorporates prior beliefs both about the true underlying function as well as the noise model. In the present study, the following exponential covariance model is used:

$$k(x_i, x_j) = e^{-(x_i - x_j)^T \Theta (x_i - x_j)} + \theta_{d+1} \tag{4.3}$$

where $\Theta = \text{diag}\{\theta_1, \theta_2, ..., \theta_d\} \in \mathbb{R}^{d \times d}$ is a diagonal matrix of undetermined hyperparameters, and $\theta_{d+1} \in \mathbb{R}$ is an additional hyperparameter arising from the assumption that noise in the dataset is Gaussian (and output dependent). Then the symbol $\boldsymbol{\theta}$ is used to denote the vector of undetermined hyperparameters, i.e., $\boldsymbol{\theta} = \{\theta_1, \theta_2, ..., \theta_{d+1}\}$.

In practice, the undetermined hyperparameters are tuned to the data using the evidence maximization framework. Once the hyperparameters have been estimated from the data, predictions can be readily made for a new testing point. To illustrate this, assume that $\mathbf{t}_n$ represents the set of n targets, $\mathbf{C}_n$ the corresponding covariance matrix and that the process to be modeled has zero mean, i.e., $\boldsymbol{\mu} = 0$. Given a new point $\mathbf{x}_{n+1}$, it can be shown that the prediction t_{n+1} has a conditional probability distribution given by :

$$P(t_{n+1}|\mathcal{D}, \mathbf{C}, \mathbf{x}_{n+1}) = \frac{1}{Z} \exp\left(-\frac{(t_{n+1} - \hat{t}_{n+1})^2}{2\hat{\sigma}^2}\right) \tag{4.4}$$

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

where,

$$\hat{t}_{n+1} = \mathbf{k}_{n+1}^T(\mathbf{x})\mathbf{C}_n^{-1}\mathbf{t}_n \tag{4.5}$$

and

$$\sigma_{n+1}^2 = \mathbf{k}(\mathbf{x}_{n+1}, \mathbf{x}_{n+1}; \boldsymbol{\theta}) - \mathbf{k}_{n+1}^T(\mathbf{x})\mathbf{C}_n^{-1}\mathbf{k}_{n+1} \tag{4.6}$$

where, $\hat{t}_{n+1}$ and σ_{n+1}^2 are the predicted posterior mean and variance, respectively, and $\mathbf{k}_{n+1} = \{k(x_{n+1}, x_1), k(x_{n+1}, x_2), \ldots, k(x_{n+1}, x_n)\} \in \mathbb{R}^n$. More specifically, $\hat{t}_{n+1}$ is the mean prediction at point $\mathbf{x}_{n+1}$, σ_{n+1} is the standard deviation of t_{n+1} and provides a measure of the confidence at point $\mathbf{x}_{n+1}$.

From a computational perspective, the search for an optimal GP approximation under the evidence maximization framework [97] involves solving the following nonlinear maximum likelihood estimation (MLE) problem to determine the most probable hyper-parameters $\boldsymbol{\theta}_{MP}$ for the given data:

$$\boldsymbol{\theta}_{MP} = \min_{\boldsymbol{\theta}} L(\boldsymbol{\theta}) \tag{4.7}$$

where

$$L(\boldsymbol{\theta}) = -\frac{1}{2}\log \Delta\mathbf{C}_n - \frac{1}{2}\mathbf{t}_n^T\mathbf{C}_n^{-1}\mathbf{t}_N - \frac{n}{2}\log 2\pi \tag{4.8}$$

is the negative log likelihood function.

The main computational cost involved in constructing GP surrogate models occurs in the MLE phase. Since computing $L(\boldsymbol{\theta})$ and its gradient generally involves computing and decomposing a dense $n \times n$ covariance matrix ($\mathcal{O}(n^3)$ operations) at each iteration, training the GP model can be prohibitively expensive even for moderately sized data (e.g., say a few thousand data points). It is worth noting that an approximation method requiring high computational cost has limited utility in a surrogate-assisted evolutionary optimization framework.

The computational bottleneck in standard GP modeling can be alleviated by employing a data parallel approach, which makes it possible to deal with datasets containing

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

tens of thousands of points at modest computational cost [5]. Since a Gaussian stochastic process is completely specified by its covariance function, training a GP involves considering a parameterized covariance function and determining its hyperparameters $\boldsymbol{\theta}$ such that the log likelihood of the data is maximized. Next, a compactly supported covariance function is outlined to facilitate data parallel GP learning.

To illustrate our approach, let us assume the existence of p disjoint and spatially localized subsets of the training data say $\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_p$. This partitioning of data can be readily achieved using the greedy load balancing clustering algorithm proposed by Choudhury *et al.* [5]. Given such a partitioning, the following covariance model can be employed to model the data.

$$\tilde{k}(\mathbf{x}_i, \mathbf{x}_j; c(\mathbf{x}_i), c(\mathbf{x}_j), \boldsymbol{\theta}) = \delta_{c(\mathbf{x}_i), c(\mathbf{x}_j)} k(\mathbf{x}_i, \mathbf{x}_j; \boldsymbol{\theta}) \tag{4.9}$$

where $\mathbf{x}_i, \mathbf{x}_j \in \mathbb{R}^d$ are input vectors, δ_{ij} is the Kronecker delta function, $\boldsymbol{\theta}$ is a set of hyperparameters and $c : c(\mathbf{x}) \mapsto 1, 2, \ldots, p$ is an assignment function which maps the input point $\mathbf{x}$ to one of p available clusters. Then the covariance function in (4.8) can be immediately written for cluster i as :

$$\begin{aligned} \tilde{k}(\mathbf{x}_1, \mathbf{x}_2; c(.), \boldsymbol{\theta}) = &\quad k(\mathbf{x}_1, \mathbf{x}_2; \boldsymbol{\theta}), \quad c(\mathbf{x}_i) = c(\mathbf{x}_j) = i \\ = &\quad 0, \qquad otherwise \end{aligned} \tag{4.10}$$

where $\boldsymbol{\theta}_i$ denotes the set of hyperparameters for the local model trained on the ith cluster. Consider the case when $p = 2$, i.e, when the data has been partitioned into two disjoint spatially localized subsets. Then using (4.10), the covariance matrix can be written as:

$$\mathbf{K} = \begin{bmatrix} \mathbf{K}_{11} & \mathbf{0} \\ \mathbf{0} & \mathbf{K}_{22} \end{bmatrix} \tag{4.11}$$

where $\mathbf{K}_{ii} \in \mathbb{R}^{n_i \times n_i}$ contains correlation terms explicitly from the ith cluster which consists of n_i points. Since in this case the determinant of the covariance matrix $\mathbf{K}$ can be

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

written as the product of determinants of the blocks $\mathbf{K}_{11}$ and $\mathbf{K}_{22}$, the log likelihood can be split into individual log likelihoods for the two partitions, i.e.,

$$L(\boldsymbol{\theta}) = L(\boldsymbol{\theta}_1) + L(\boldsymbol{\theta}_2). \tag{4.12}$$

From the preceding discussion, it is clear that the use of a compactly supported covariance function naturally leads to a data parallel learning approach to GP approximation and hence provides a means to handle large datasets. In general, it is often the case that the predictive capability may reduce when an increasing number of clusters are used [5]. However, this degradation in performance is often very small and acceptable given the significant savings in computational cost.

4.2.2 Polynomial Regression Modeling Technique

PR models can easily be fitted to data with a least squares approach. Here, the PR modeling technique is described for approximating multi-dimensional input data of any order [98].

Let $\mathcal{D} = \{\mathbf{x}_i, t_i\}, i = 1 \ldots n$ denote the training dataset, where $\mathbf{x}_i \in \mathbb{R}^d$ is the input design vector, $t_i \in \mathbb{R}$ is the corresponding target value, and $\mathbf{x}_i = (x_{i_1}, x_{i_2}, \ldots, x_{i_d})$, d denotes the dimension of the problem, then:

$$t_i = f(\mathbf{x}_i) = f(x_{i_1}, \ldots, x_{i_d}) \tag{4.13}$$

Define an exponent vector ε containing positive integers $(\pi_1, \pi_2, \ldots, \pi_d)$ and define $\mathbf{x}_i^{\varepsilon}$ as an exponent input vector $({x_{i_1}}^{\pi_1}, {x_{i_2}}^{\pi_2}, \ldots, {x_{i_d}}^{\pi_d})$.

Given a set of exponent vectors $\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_m$ and the set of data $(\mathbf{x}_i, t_i)$, where $i = 1, 2, \ldots, n$, the polynomial model of $(m-1)^{th}$ order has the form:

$$\hat{t}_i = C_1 \mathbf{x}_i^{\varepsilon_1} + C_2 \mathbf{x}_i^{\varepsilon_2} + \ldots + C_m \mathbf{x}_i^{\varepsilon_m} \tag{4.14}$$

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

where $C_1, C_2, \ldots, C_m$ are the coefficient vectors to be estimated, $C_j = (c_{j_1}, c_{j_2}, \ldots, c_{j_d})$, $j = 1, 2, \ldots, m$.

The least square method is then used to estimate the coefficients of the polynomial model. By definition, the least square error E to be minimized is:

$$E = \sum_{i=1}^{n} [t_i - \hat{t}_i]^2 \tag{4.15}$$

It may be easily shown that $t_i = f(\mathbf{x}_i)$. By multiplying both sides of (4.14) with $\mathbf{x}_i^{\varepsilon_j}$ and taking the sum of n pairs of input-output data:

$$C_1 \sum_i \mathbf{x}_i^{\varepsilon_1+\varepsilon_j} + \ldots + C_m \sum_i \mathbf{x}_i^{\varepsilon_m+\varepsilon_j} = \sum_i t_i \mathbf{x}_i^{\varepsilon_j} \tag{4.16}$$

For $j = 1, 2, \ldots, m$, the polynomial model for the training dataset can be represented in the matrix notation as follows

$$A\gamma^T = \boldsymbol{b}^T \tag{4.17}$$

where

$$A = \begin{bmatrix} \sum_i \mathbf{x}_i^{\varepsilon_1+\varepsilon_1} & \ldots & \sum_i \mathbf{x}_i^{\varepsilon_1+\varepsilon_m} \\ \vdots & & \vdots \\ \sum_i \mathbf{x}_i^{\varepsilon_m+\varepsilon_1} & \ldots & \sum_i \mathbf{x}_i^{\varepsilon_m+\varepsilon_m} \end{bmatrix} \tag{4.18}$$

$$\boldsymbol{b} = (\sum t_i \mathbf{x}_i^{\varepsilon_1}, \ \ldots \ , \sum t_i \mathbf{x}_i^{\varepsilon_m}) \tag{4.19}$$

$$\gamma = (C_1, C_2, \ldots, C_m) \tag{4.20}$$

Then the coefficient matrix of the polynomial is:

$$\gamma = (A^{-1}\boldsymbol{b}^T)^T \tag{4.21}$$

Considering $B_i = (\mathbf{x}_i^{\varepsilon_1}, \ldots, \mathbf{x}_i^{\varepsilon_m})$, the following equations may be derived:

- $A = \sum_i B_i^T B_i$

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

- $\boldsymbol{b} = \sum_i t_i B_i$

The predicted output for a new input pattern is then given by $\hat{t}_i = \gamma.B_i^T$.

In optimization, the smoothing capability of polynomial regression allows quick convergence of noisy functions. The major advantage of PR model is also the relatively lower computational efforts consumed in model building. However, PR model may not approximate complex landscape sufficiently well or accurately.

4.2.3 Pre-Selection Criterion

The pre-selection criterion in HSAMA serves to filter out the most promising individuals that will undergo the local search strategy. That is to say, this statistical criterion is only used to filter the individuals in an EA population. A local search strategy is then employed to identify the best solution in the vicinity of an individual. Here, the pre-selection criteria used in PR-HSAMA and GP-HSAMA are introduced.

4.2.3.1 Mean of Model Prediction

An obvious and common pre-selection criterion is to use the predicted fitness value $\hat{t}$ provided by the global surrogate model to find the promising individuals, which is referred as Mean of Model Prediction (MMP) pre-selection criterion here. This MMP pre-selection criterion is used for the empirical study of the PR global surrogate model based HSAMA, i.e., PR-HSAMA. However, this criterion may lead to premature convergence in many cases due to the inevitable limitations on the accuracy of a global surrogate model constructed using a few data points [65]. Hence, there is also a need to explore new areas of search space for a more thorough global search, such as the probability of improvement.

4.2.3.2 Probability of Improvement

As mentioned earlier, the GP approach results in a random field approximation of the analysis code. Using the output mean prediction $\hat{t} = \hat{f}(\mathbf{x})$ and standard deviation $\sigma(\mathbf{x})$

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

of GP model, a variety of pre-selection criteria for the selection of promising individuals may be formulated to accelerate evolutionary optimization search. Here the Probability of Improvement (PoI) is considered for the empirical study of GP-HSAMA since previous work in [65] suggests that it performs well.

To illustrate the PoI approach, consider the case when it is aimed to solve a minimization problem. Let t^- denote the smallest value of all the outputs in the training dataset used to construct the GP surrogate. Subsequently, it is intended to use the surrogate model to predict a new point $\mathbf{x}^*$ at which the output is likely to be lower than t^-. The PoI at the point $\mathbf{x}^*$ (i.e., the probability that the surrogate prediction at $\mathbf{x}^*$ is lower than t^-) can be readily computed from the posterior mean $\hat{f}(\mathbf{x}^*)$ and standard deviation $\sigma(\mathbf{x}^*)$ as follows:

$$PoI(\mathbf{x}^*) = \Phi\left(\frac{t^- - \hat{f}(\mathbf{x}^*)}{\sigma(\mathbf{x}^*)}\right) \tag{4.22}$$

where $\Phi(.)$ is the normal cumulative distribution function.

Figure 4.1 shows the characteristics of the PoI pre-selection criterion for a one-dimensional test function. It may be noted from the figure that the PoI criterion is able to correctly identify the region in which the true objective function must be sampled to drive $f(x)$ below t^-. The points identified by maximizing $PoI(\mathbf{x})$ can be appended to the baseline training dataset to update the surrogate model (and consequently the PoI criterion). Increasing the number of training points in such a stage-wise fashion improves the ability of the PoI criterion to correctly locate the region in which the optimum lies. The possibility of employing alternative statistical measures such as the expected improvement criterion proposed by Jones *et al.* [38].

4.3 Performance Analysis of GP-HSAMA

In this section, the performance of the proposed HSAMA evolutionary optimization framework is analyzed by using the GP global model, i.e., GP-HSAMA. The performance

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

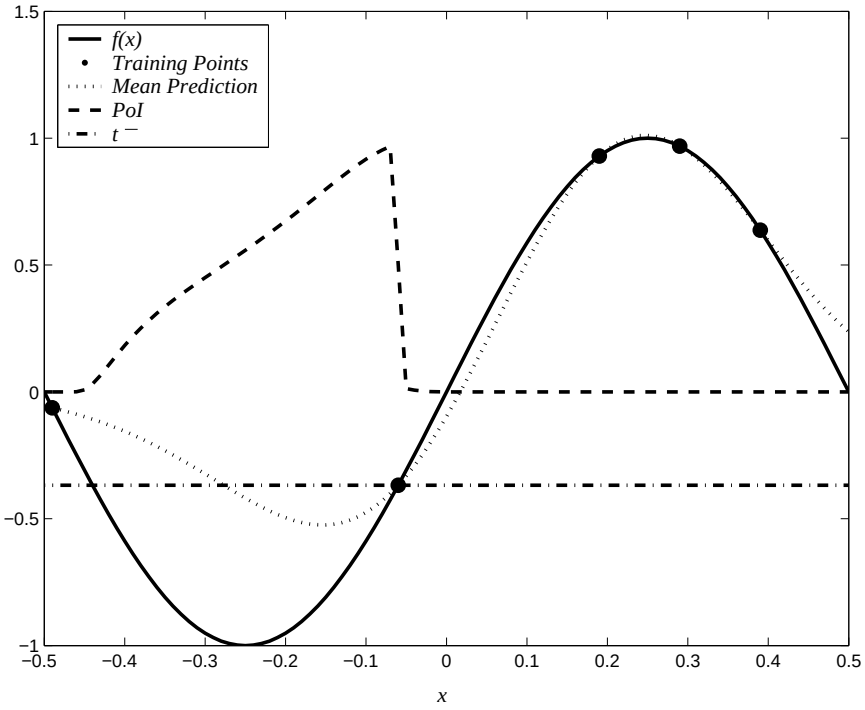


Figure 4.1: Characteristics of the *PoI* criterion when the GP model is trained on points generated by a one-dimensional function.

of the HSAMA algorithm is evaluated against a standard GA, and two surrogate-assisted evolutionary optimization algorithms that were recently introduced in the literature [6] and [8]. One is the representative of Surrogate-Assisted Evolutionary Algorithm with Global-search Strategy, referred as SAGA-GS since the GA is used, the other is the representative of Surrogate-Assisted Evolutionary Algorithm with Local-search Strategy, i.e., SAMA-TRF.

At each search generation, the SAGA-GS employs the standard GP surrogate model to screen the entire population of individuals. The predefined top ranking $\eta\%$ individuals in the EA population then undergo exact evaluations. In contrast to [6], the SAGA-GS employed in our study involves using the computationally cheap DPGP and estimates the ranking of the individuals based on their probability of improvements rather than merely using the mean prediction. On the other hand, the SAMA-TRF considered corresponds

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

PARAMETER	VALUE
population size	50
crossover probability	0.6
mutation rate	0.001
number of nearest neighboring data points M	100
maximum trust region iterations k_{max}	3
max number of design points for global model	2000
pre-selection percentage η	40%

Table 4.1: The configurations of GA, SAGA-GS, SAMA-TRF and HSAMA used in empirical study.

to the earlier work in [8] that evolves the solution of each individual using Lamarckian learning based on local RBF surrogates.

A standard GA is employed with population size of 50, uniform crossover and mutation operators at probabilities 0.6 and 0.001, respectively. However, apart from the standard GA settings, the two user-specified parameters of the SAMA-TRF are – (1) number of nearest neighboring data points used to construct the local surrogate model M and (2) maximum trust region iterations k_{max}. In present numerical studies, the M and k_{max} is set to 100 and 3, respectively. In SAGA-GS and HSAMA, the maximum number of training design points and clusters for constructing the global surrogate model using DPGP are configured as 2000 and 4, respectively. While η is configured as 40%. In addition, all configurations used in this study were values suggested in earlier studies [5], [6], [8]. The reader is referred to Table 4.1 about the configurations of GA, SAGA-GS, SAMA-TRF and HSAMA used in empirical study.

The results obtained from the empirical studies on a range of benchmark test functions, i.e., two unimodal test functions (Sphere and Rosenbrock test function) and three multimodal test functions (Ackley, Griewank and Rastrigin test function) are presented in Figs. 4.2-4.6. All benchmark test functions used in the study are of 20 dimensions and have a single global minimum at zero (see the appendix A for greater details of the

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

test functions). It should be noted that all results presented are average of 20 simulations conducted with a tractable computational budget of (6×10^3) exact fitness function evaluations.

In the context of surrogate-assisted evolutionary search, it is common practice to limit the function evaluation to some low values. In my dissertation work, the proposed SAEAs are demonstrated in the context of aerodynamic shape design design optimization where a single objective function (see equation (B.2)) evaluation takes approximately 20 minutes to compute using a Pentium IV 1.9GHZ workstation. During the preliminary design phase, the amount of time that designer has is often limited and usually does not exceed 1 or 2 months. By considering the function evaluations of $3,000$ and $6,000$ imply a wall-clock time of $3,000 * 20 \approx 1.5$ months and $6,000 * 20 \approx 3$ months respectively. As a result, limiting function evaluations to the order of 3,000 or 6,000 in most simulations of my research work is a realistic choice.

From the results obtained in Figs. 4.2-4.6, it is clear that all the surrogate-assisted evolutionary optimization algorithms considered here are capable of searching more efficiently than the standard GA on the benchmark problems under a tractable computational budget. Further, both SAMA-TRF and GP-HSAMA appear to converge much faster and yield improved solution quality as compared to SAGA-GS on all the benchmark problems. This makes sense since Memetic algorithms, i.e., EAs that employ local search heavily such as SAMA-TRF and GP-HSAMA, are generally well-known to search more effectively and efficiently. The superiority of SAMA-TRF and GP-HSAMA are more evident on the unimodal benchmark problems.

It is worth noting that GP-HSAMA converges significantly faster than the SAMA-TRF on the unimodal problems. For instance, the GP-HSAMA converges correctly to the global minimum of the exact objective function in Fig. 4.2 within the tractable computational budget. This outcome may be easily explained. Since the Sphere problem is

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

a smooth, symmetric function and unimodal, it makes perfect sense to use the Lamarckian learning process in GP-HSAMA or SAMA-TRF involving any gradient-based local search. However, in contrast to SAMA-TRF, only the $\eta\%$ top ranking individuals among the entire EA population in GP-HSAMA undergo the Lamarckian learning process, thus providing significant computational cost savings.

Consider next the complex multimodal benchmark problems. On multimodal functions, the number of local minima increases exponentially with the problem dimensions, often they present hills and valleys with misleading local optima. Any gradient-based optimization algorithm would easily get stuck in a local minimum. Hence, performance studies of surrogate-assisted EAs on multimodal problems reflect the algorithm's ability to escape from poor local optima and head towards the global optimum. Figures 4.4-4.6 illustrate the search performances of GA, SAGA-GS, SAMA-TRF and GP-HSAMA on the Ackley, Griewank and Rastrigin multimodal benchmark test functions, respectively. From these figures, GP-HSAMA is once again demonstrated to accelerate the evolutionary search significantly faster than GA, SAGA-GS or SAMA-TRF on all of the multimodal problems considered. For the Ackley function, the GP-HSAMA is capable of converging correctly to the global minimum of the exact objective function even though there are thousands of local minima in the entire search space (see Fig. 4.4). This indicates the robustness of the GP-HSAMA to prevent premature convergence.

Overall, the results obtained also imply that the GP-HSAMA is not only capable of identifying the better quality individuals in each EA population (via its global search strategy), at the same time, its intrinsic local search strategy can also exploit these filtered individuals effectively and efficiently. This combination of the global and local search strategies in the HSAMA is the key reason for the improvements in search quality at a significantly lower computational budget than existing surrogate-assisted EAs.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

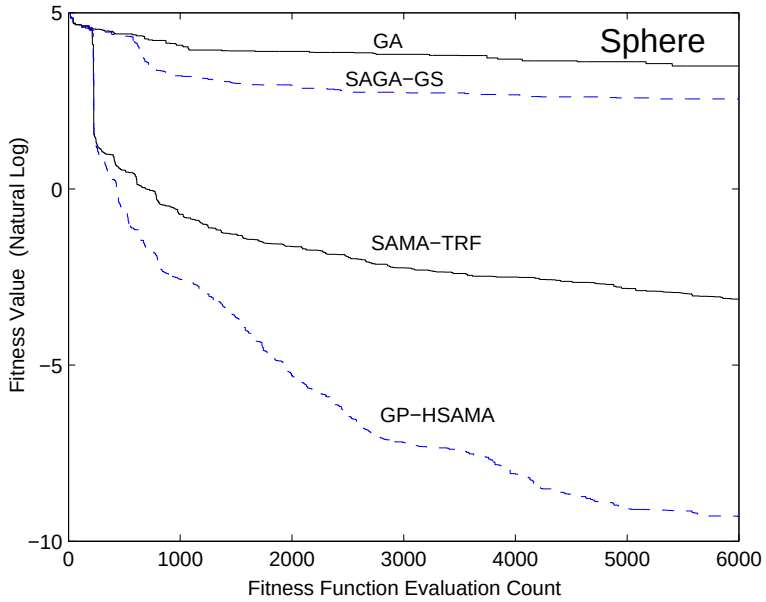


Figure 4.2: Convergence trends of the GA, SAGA-GS, SAMA-TRF and GP-HSAMA framework for Sphere function.

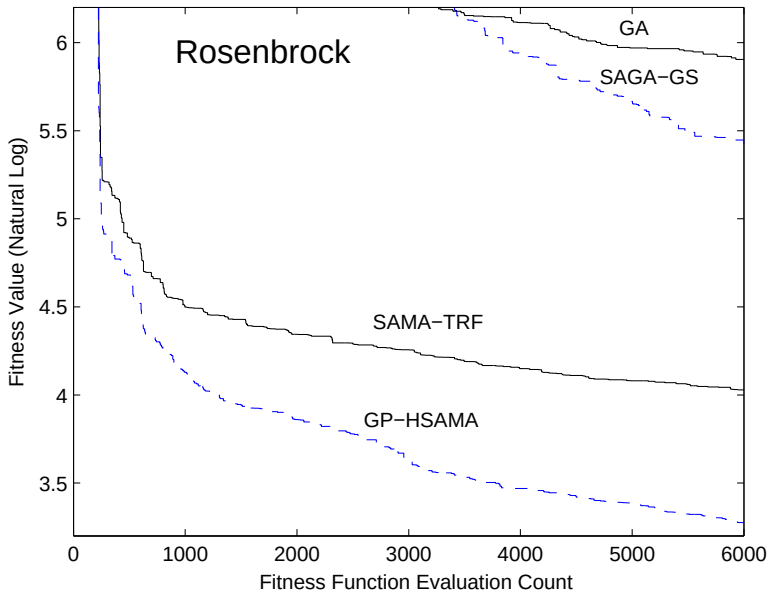


Figure 4.3: Convergence trends of the GA, SAGA-GS, SAMA-TRF and GP-HSAMA framework for Rosenbrock function.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

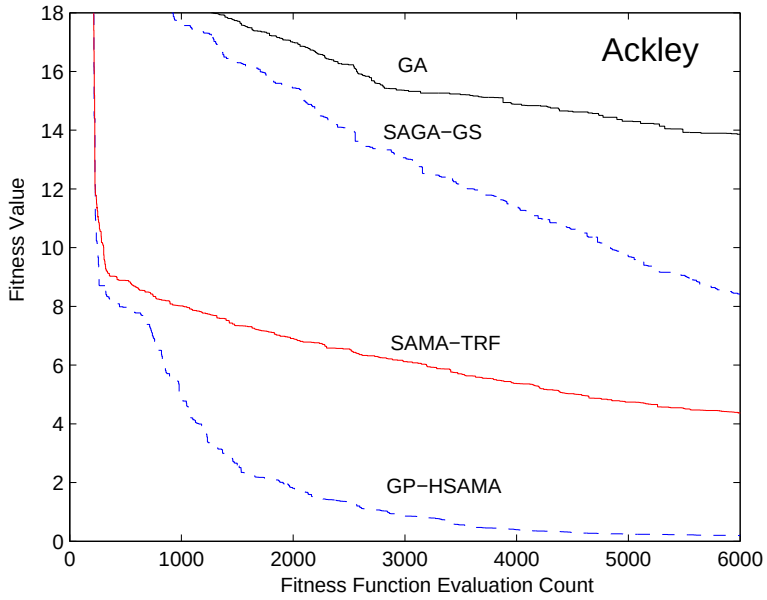


Figure 4.4: Convergence trends of the GA, SAGA-GS, SAMA-TRF and GP-HSAMA framework for Ackley function.

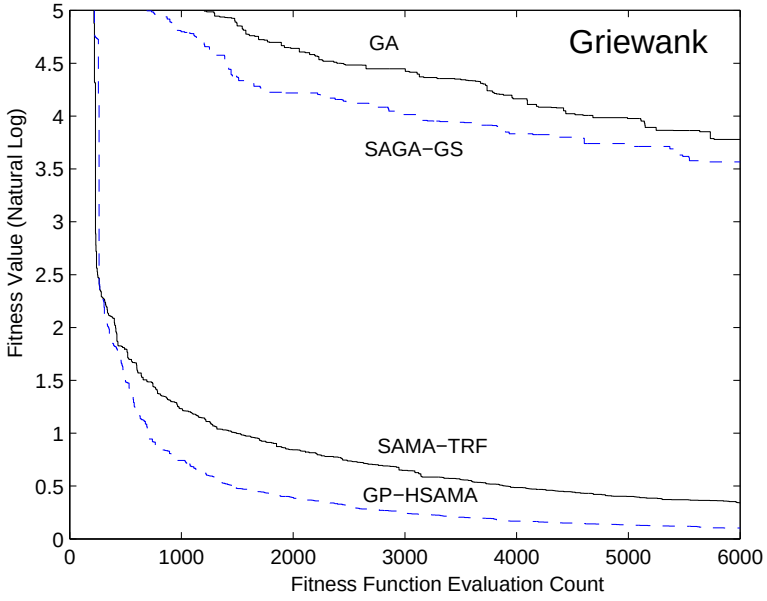


Figure 4.5: Convergence trends of the GA, SAGA-GS, SAMA-TRF and GP-HSAMA framework for Griewank function.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

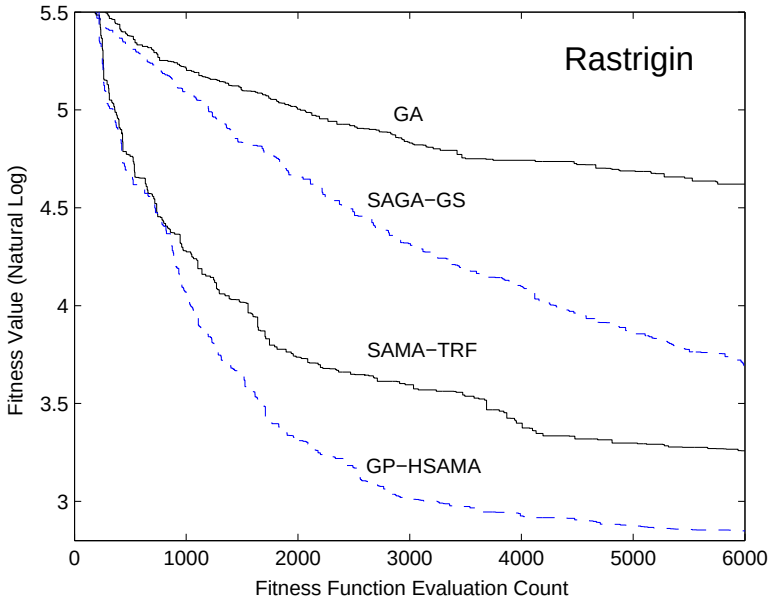


Figure 4.6: Convergence trends of the GA, SAGA-GS, SAMA-TRF and GP-HSAMA framework for Rastrigin function.

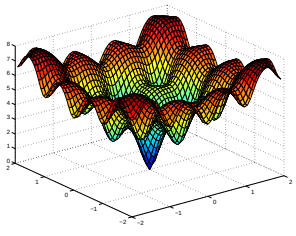
4.4 Comparative Study of HSAMA Variants

In this section, a further comparative study of several HSAMA variants is presented on Ackley and Griewank multi-modal benchmark test functions so as to investigate the convergence properties of the HSAMA using dissimilar global surrogate models. The HSAMA using PR or DPGP global surrogate model is referred in short as PR-HSAMA or GP-HSAMA, respectively. Further, the parameter configurations are kept in consistent with our previous studies on performance analysis of the HSAMA in Section 4.3.

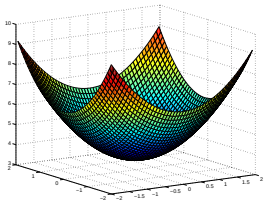
4.4.1 Effect of Different Order of PR Model

Next, the effect of different order of PR model is presented for approximating the multi-modal Ackley function.

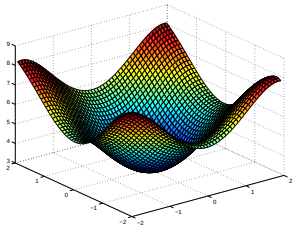
ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library



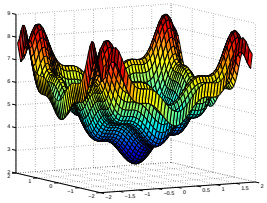
4.7.a: 2-D Ackley Function.



4.7.b: 2nd order PR Approximation.



4.7.c: 5th order PR Approximation.



4.7.d: 10th order PR Approximation.

Figure 4.7: 2-D Ackley function and its PR approximation models.

The two dimensional Ackley function depicted in Figure 4.7.a is approximated using 2nd, 5th and 10th order polynomial regression and 300 training data points, which are plotted in Figures 4.7.b to 4.7.d, respectively. The Ackley function is multi-modal with many local minima and a global minimum located at (0,...,0). From the figures, it may be observed the 2nd order PR model is unable to approximate the complex landscape of the Ackley function very well. The approximated Ackley function becomes a quadratic function. Higher order PR models, on the other hand, approximate the multi-modal landscape of the Ackley function more closely. However, a higher order PR incurs greater computational cost to construct the global model. The main computational cost of PR modeling is $\mathcal{O}((dm)^3)$ for n pairs of data, and $(m-1)^{th}$ order, where d denotes the dimension of the problem and m denotes the order number of PR model

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

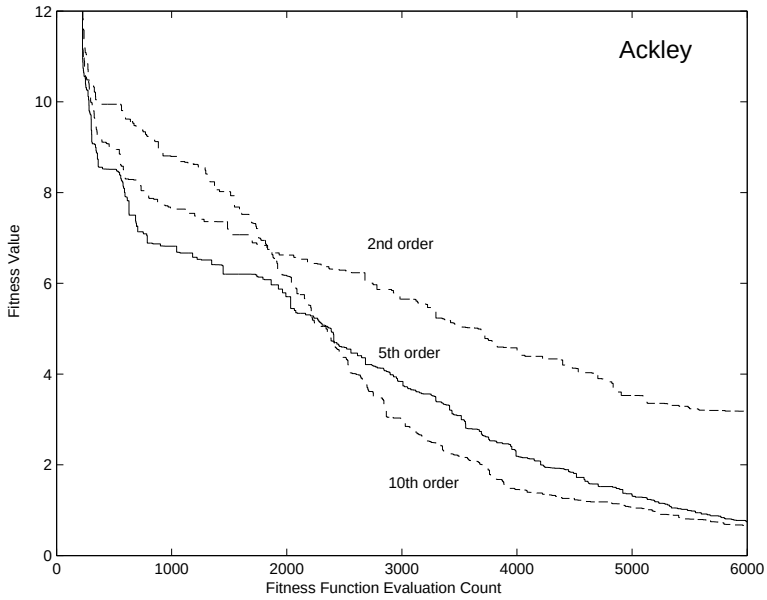


Figure 4.8: Averaged search trends of PR-HSAMA for different orders on 20-dimensional Ackley function.

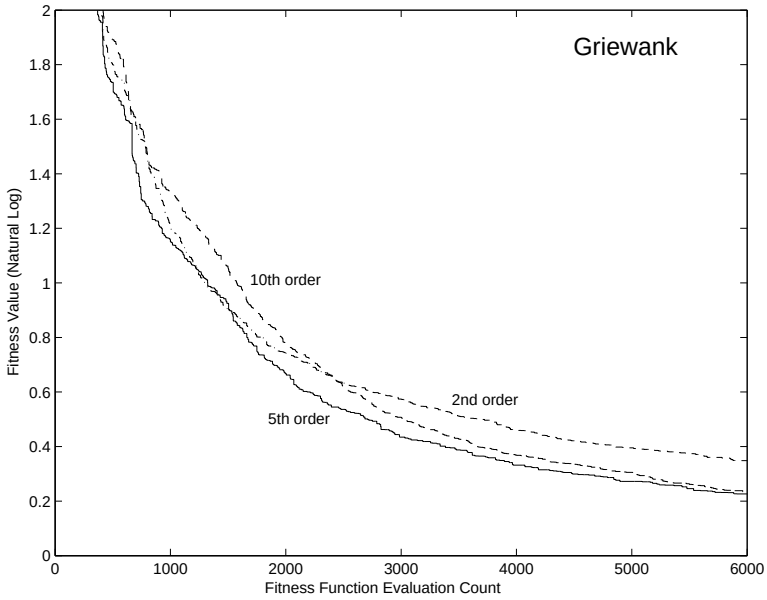


Figure 4.9: Averaged search trends of PR-HSAMA for different orders on 20-dimensional Griewank function.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

The convergence behavior of HSAMA framework using different orders of PR for global surrogate modeling is also studied. Figures 4.8 and 4.9 depict the search traces of PR-HSAMA search on the multimodal Ackley and Griewank multi-modal test functions. The PR^{2nd}-HSAMA fares poorer than both the 5th and 10th order counterparts on the test functions. This may be attributed to the inferior approximation accuracy of the 2nd order PR global model which results in poor identification of promising individuals during the pre-screening process.

The results in figures 4.8 and 4.9 also show little improvements in using the PR^{10th}-HSAMA over the 5th order counterpart. In effect, the PR^{5th}-HSAMA is used in subsequent experimental study since it is more computationally efficient, but provides solution quality that is competitive to the PR^{10th}-HSAMA.

4.4.2 Comparison of PR- and GP-HSAMAs

Figures 4.10 and 4.11 depicts the search performance traces of the $5th$ order PR-HSAMA and GP-HSAMA on the Ackley and Griewank functions, respectively.

It may be observed in these figures that both PR-HSAMA and GP-HSAMA were able to filter off unnecessary local searches, thus accelerating convergence and providing significant computational cost savings over the SAMA-TRF. Overall, GP-HSAMA converges to better designs faster than PR-HSAMA. It is observed that this is due to the inability of the PR method to approximate the complex high dimensional landscape accurately, resulting in poor qualitative fitness approximation. Consequently, the PR global surrogate model fails to accurately identify the $\eta\%$ top ranking promising individuals in the EA population that will undergo Lamarckian learning.

It is worth noting that PR-HSAMA is still capable of providing significant computational cost savings over the SAMA-TRF. This is most likely due to the second level surrogate-assisted local search which helps maintain good accuracy in the evolutionary search.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

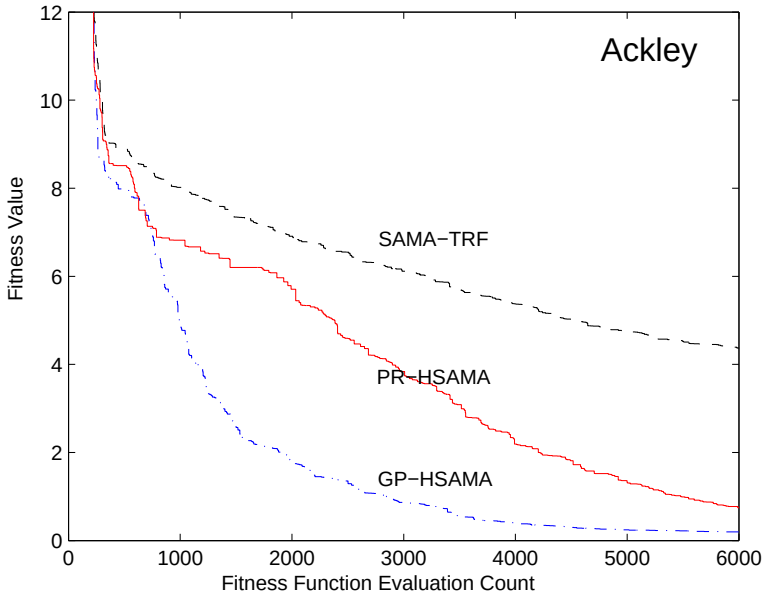


Figure 4.10: Averaged search trends of SAMA-TRF, GP-HSAMA, and PR-HSAMA on 20-dimensional Ackley function.

4.4.3 Wall Clock Time of PR- and GP-HSAMAs

Many existing works on surrogate-assisted evolutionary algorithms only report the fitness values obtained against the number of exact evaluation calls made to the computationally expensive optimization problem. However, it is obvious there is significant difference in the computationally efforts incurred by GP and PR for surrogate modeling. Hence, it would be interesting to consider the actual wall clock time of the PR-HSAMA and GP-HSAMA, inclusive of the modeling time. The computational efforts for a single generation of the HSAMA search may be formulated as

$$T_{gen} = T_g + T_{ls} + k \cdot \eta \cdot n_{pop} \cdot T_{obj} + T_{ga} \tag{4.23}$$

where T_g, T_{ls}, T_{obj} and T_{ga} denote the wall clock time to construct the global surrogate model, complete a trust-region local search with on-line RBF local surrogate models,

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

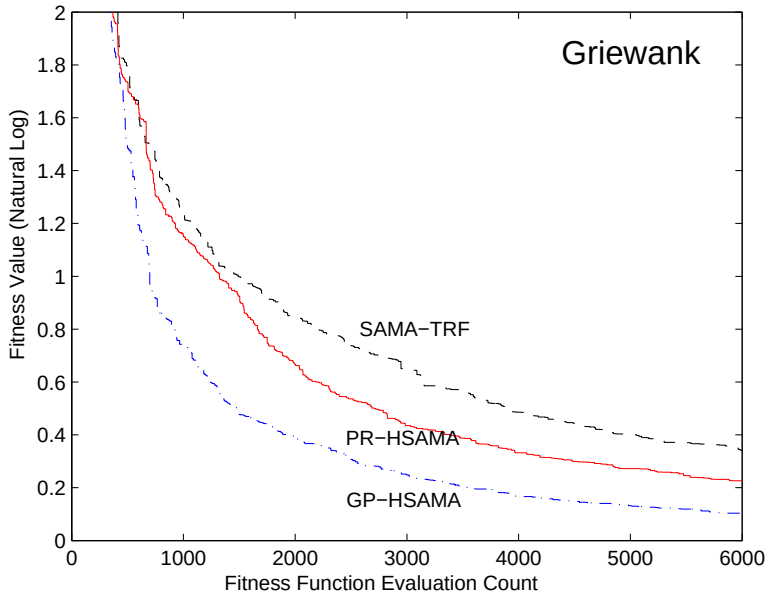


Figure 4.11: Averaged search trends of SAMA-TRF, GP-HSAMA, and PR-HSAMA on 20-dimensional Griewank function.

evaluate a single exact objective function and perform standard GA operators, and n_{pop} is the population size.

In solving optimization problems, the terms T_{ls}, k, η, n_{pop} and T_{ga} in equation (4.23) are the same for both PR-HSAMA and GP-HSAMA, but differs in the time taken to constructing the global surrogate model, T_g. The main computational cost involved in constructing GP surrogate models occurs in the maximum likelihood estimation phase so as to determine the most probable hyperparameters [97]. The evaluation of the likelihood function requires factorization of the correlation matrix and scales as $\mathcal{O}(n^3)$. In comparison, the main computational cost in constructing PR model is performing matrix inversion which scales as $\mathcal{O}((dm)^3)$ but is independent on the number of training data, i.e. n in GP model.

Since the number of training data, n, used to built the models is often large in size and $dm \ll n$, the time to construct PR modeling is significantly lower compared

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Model Type	Using training data size of	
	500	2000
the 5th order PR	0.333165	1.20895
DPGP	11.912108	1255.4707

Table 4.2: Average wall clock time (seconds) to construct surrogate model

to using GP. Table 4.2 tabulates the average wall clock time to construct the GP and PR global surrogate models for different sizes of training dataset using a Pentium IV 1.9GHZ workstation. It should be noted that the training dataset is generated using Latin hypercube sampling.

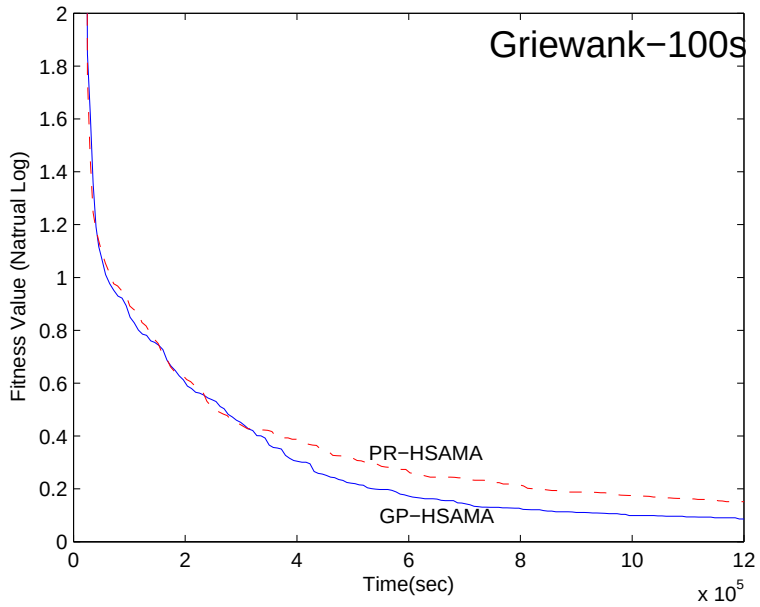


Figure 4.12: Wall clock time of GP-HSAMA and PR-HSAMA on the 20-Dimensional 100s Griewank Function.

With the great difference in the time taken to construct models using PR and GP, it is apparent some approximation methods may not be efficient if the the computational expense of the optimization problem is not very high. Hence, it would be interesting to investigate a suitable approximation method for the optimization problem in hand.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

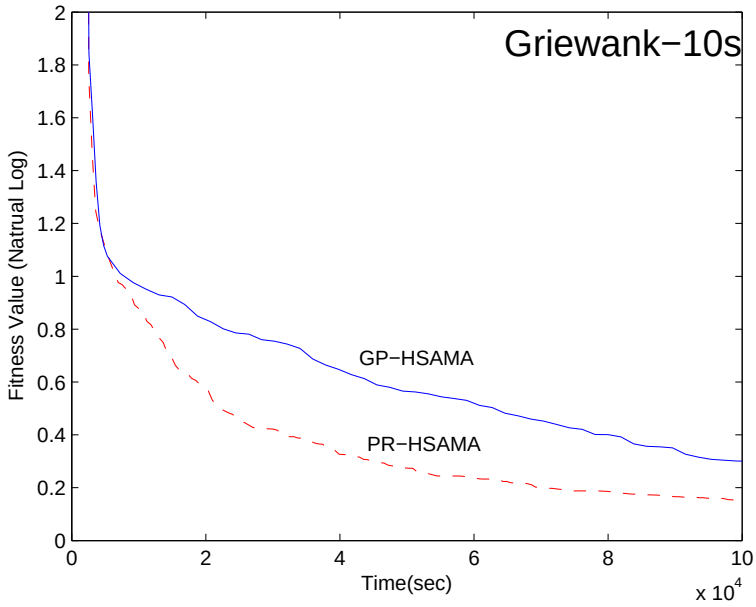


Figure 4.13: Wall clock time of GP-HSAMA and PR-HSAMA on the 20-Dimensional $10s$ Griewank Function.

From equation (4.23), T_{ls} and T_{ga} are the same for both PR-HSAMA and GP-HSAMA, while the key difference is T_g which is the wall clock time to construct the global surrogate model. As a result, when using HSAMA to solve an optimization problem having a computational expense of T_{obj}, the following condition can be derived when one should use a GP global model over PR:

$$T_{obj} \gg \frac{T_{gp}}{k \cdot \eta \cdot n_{pop}} \tag{4.24}$$

where T_{gp} is the wall clock time to construct a GP model in each EA generation, and n_{pop} is the population size.

On the other hand, the PR method is preferred over GP when:

$$\frac{T_{pr}}{k \cdot \eta \cdot n_{pop}} \ll T_{obj} < \frac{T_{gp}}{k \cdot \eta \cdot n_{pop}} \tag{4.25}$$

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

where T_{pr} and T_{gp} are the wall clock time to construct a PR or GP model in each EA generation, respectively.

To validate these conditions derived in equations (4.24) and (4.25), the search performances of HSAMA for optimization problems with different computation time of fitness evaluation are investigated empirically by using the Griewank function. Note that in the present experiment, all parameter configurations are kept the same as Table 4.1 in Section 4.3. By substituting the values of k, η, n_{pop}, T_{pr} and T_{gp} in equations (4.24) and (4.25), then $T_{obj} \gg 20.924512$ and $0.0201492 \ll T_{obj} < 20.924512$, respectively.

Figures 4.12 and 4.13 show the search traces of GP-HSAMA and PR-HSAMA on the $100s$ and $10s$ Griewank function (i.e., a single fitness evaluation time of the Griewank function is 100 seconds or 10 seconds). On the $100s$ Griewank function, i.e., $(T_{obj} = 100) \gg 20.924512$, GP-HSAMA is observed to perform better than PR-HSAMA. On the other hand, PR-HSAMA is superior to GP-HSAMA on the $10s$ Griewank function when $0.0201492 \ll (T_{obj} = 10) < 20.924512$.

4.5 Aerodynamic Shape Design Optimization

In the real application of aerodynamic shape design area, each function evaluation may require the numerical solution of the Euler or Navier-stokes equations, which often take up many minutes to hours of CPU time. An optimization method that requires many thousands of fitness calls to a computationally expensive simulation code has limited usefulness in such design problems. This motivates the application of the proposed HSAMA to aerodynamic shape design.

In this section, the proposed HSAMA is applied to efficient aerodynamic shape design. In particular, the parametric design optimization of a 2-D airfoil structure with minimum drag-over-lift ratio, i.e. D/L (see fitness function of equation (B.2)), is considered in

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

PARAMETER	VALUE
population size	20
crossover probability	0.6
mutation rate	0.001
number of nearest neighboring data points M	60
maximum trust region iterations k_{max}	3
pre-selection percentage η	40%

Table 4.3: The configurations of GA, SAMA-TRF, HSAMA for aerodynamic shape design

present study. The reader is referred to Appendix B for greater details of the aerodynamic shape design optimization problem used.

For the airfoil problem, a single exact computational fluid dynamics (CFD) simulation takes many minutes to compute. When dealing with computationally expensive problems that cost a long CPU time per function evaluation, surrogate model training cost may be regarded as insignificant. The parametric design of the airfoil is conducted by using three evolutionary optimization frameworks, i.e., standard GA, SAMA-TRF and GP-HSAMA. It is worth noting that SAGA-GS algorithm and PR-HSAMA was omitted for the sake of brevity since it has been shown as inferior to SAMA-TRF and GP-HSAMA respectively. The reader is referred to Table 4.3 for the configurations of GA, SAMA-TRF, HSAMA for aerodynamic shape design.

The design histories of the aerodynamic 2-D airfoil optimization problem using standard GA, SAMA-TRF and GP-HSAMA frameworks are presented in Fig. 4.14. Using a population of 20 initial design points based on Latin hypercube sampling [89], these designs are evaluated using the exact CFD analysis code. All three EA frameworks proceed with the standard GA operations using the exact CFD analysis code for the first three generations. Hence, they share the same search history at the initial search phase. This initial phase represents the period where the SAMA-TRF and GP-HSAMA forms its database of past design points for constructing surrogate models later during search.

From the results obtained in Fig. 4.14, both GP-HSAMA and SAMA-TRF arrived at

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

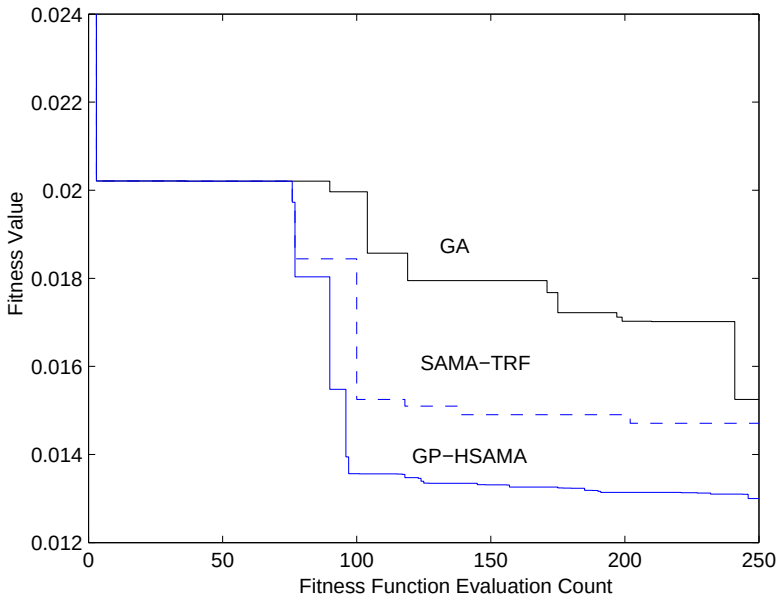


Figure 4.14: Convergence trends of the GA, SAMA-TRF and GP-HSAMA framework for aerodynamic shape design problem.

better airfoil designs than the standard GA, while incurring significantly lower computational costs. Moreover, GP-HSAMA was shown to accelerate the evolutionary search much faster as compared to both standard GA and SAMA-TRF, producing improved design much earlier.

4.6 Summary

For computationally expensive optimization problems, the use of surrogate model helps to greatly reduce the number of exact fitness evaluations by exploiting the information contained in the search history. In this chapter, a novel HSAMA optimization framework that combines both global and local surrogate models is presented. The algorithm makes use of the global surrogate model and a pre-selection criterion to rank the promising individuals in the EA population. A local surrogate-assisted Lamarckian learning approach is then applied to these promising individuals to accelerate evolutionary search.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Experimental studies are presented for a number of unimodal and multimodal benchmark test functions. The result shows that both PR-HSAMA and GP-HSAMA converge to good designs under a tractable computational budget. Further, it may be inferred that the GP method should generally be used in the HSAMA for global surrogate modeling if the evaluation function is computationally very expensive. On moderately expensive problems, the PR method may serve to be a better choice. Empirical studies are also presented for a real-world aerodynamic shape design problem. The results obtained suggest that the proposed optimization framework is capable of solving computationally expensive optimization problems more efficiently than the standard GA, SAGA-GS and SAMA-TRF under a tractable computational budget.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Chapter 5

Multi-Surrogates Assisted Memetic Algorithm

In this chapter, the notion of '*curse and blessing of uncertainty*' in approximation model is presented. Subsequently, a novel Multi-Surrogates Assisted Memetic Algorithm (MSAMA) that uses multiple surrogate modeling techniques instead of only one technique during local search phase is proposed to benefit from the effects of '*blessing of uncertainty*' and '*curse of uncertainty*' in the evolutionary optimization. In contrast to the standard SAMA, the essential backbone of our proposed MSAMA framework is an evolutionary algorithm coupled with local solvers that employ multi-surrogates. Particularly, both regressing and exact interpolating surrogates are used in the Lamarckian learning process.

5.1 Impact of Uncertainty

In this section, the impact of uncertainty, i.e., '*curse and blessing of uncertainty*', is illustrated and demonstrated by introducing surrogate model on the standard SAMA.

Surrogates models can be used to yield insights into the functional relationship between the input $\mathbf{x}$ and the output t of the training data. If the exact fitness function is represented as

$$t = f(\mathbf{x}), \tag{5.1}$$

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

then a surrogate model of $\hat{f}$ is an approximation of the form

$$\hat{t} = \hat{f}(\mathbf{x}), \tag{5.2}$$

such that $t = \hat{t} + e$, where e represents the approximation error.

5.1.1 Impact of Uncertainty on SAMA

Here, the effects of uncertainty are illustrated by introducing inaccurate surrogates on SAMA search. If $f(\mathbf{x})$ denotes the original fitness function and the approximated function is $\hat{f}(\mathbf{x})$, the approximation errors at any design point $\mathbf{x}_i$ is $e(\mathbf{x}_i)$, i.e., the uncertainty introduced by the surrogate at $\mathbf{x}_i$, may then be defined as

$$e(\mathbf{x}_i) = |f(\mathbf{x}_i) - \hat{f}(\mathbf{x}_i)| \tag{5.3}$$

For each non-duplicated individual, if n fitness calls to $\hat{f}(\mathbf{x})$ are made in the SAMA local search strategy, the root mean square error, $rmse$ can be derived as

$$rmse = \sqrt{\frac{\sum_{i=1}^{n} e^2(\mathbf{x}_i)}{n}}. \tag{5.4}$$

5.1.1.1 Curse of Uncertainty

The negative impact of '*curse of uncertainty*' on SAMA search can be illustrated using Fig. 5.1. The solid and dotted lines depicted in Fig. 5.1 denote the original one dimensional function, $f(\mathbf{x})$ and approximated function, $\hat{f}(\mathbf{x})$, respectively, assuming only 3 sparse points are available as the training dataset. Note that this is equivalent to hundreds of training data points in a multi-variate problem due to the fact that the number of hypercubes required to fill out a compact region of a d-dimensional space grows exponentially with d.

If the start point of a gradient-based local solver in SAMA is $\mathbf{x}_c$, where $f(\mathbf{x}_c) = \hat{f}(\mathbf{x}_c)$, the local search is likely to converge to the local optimum of the approximated function

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

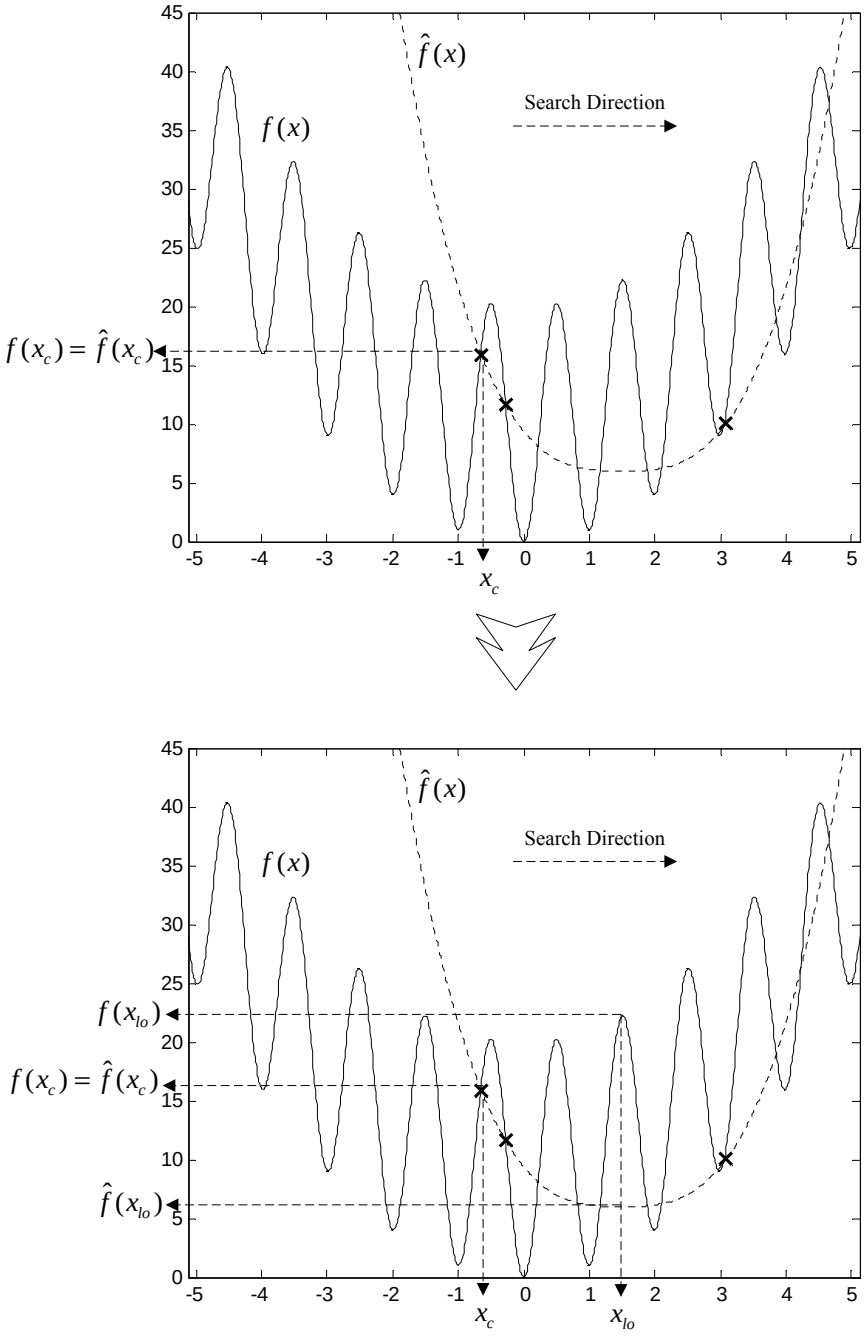


Figure 5.1: '*Curse of uncertainty*' in surrogate-assisted memetic algorithm

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

situated at $\mathbf{x}_{lo}$ which has a predicted fitness value of $\hat{f}(\mathbf{x}_{lo})$. It is worth noting that when $\mathbf{x}_{lo}$ is mapped onto the exact fitness function, there is no fitness improvement over the start point of $\mathbf{x}_c$ since $f(\mathbf{x}_{lo}) > \hat{f}(\mathbf{x}_{lo})$ and $f(\mathbf{x}_{lo}) > f(\mathbf{x}_c)$, see Fig. 5.1.

5.1.1.2 Blessing of Uncertainty

Next, the '*blessing of uncertainty*' that refers to the benefits attributed by inaccurate surrogates on SAMA search is illustrated.

The solid and dotted lines in Fig. 5.2 denote $f(\mathbf{x})$ and $\hat{f}(\mathbf{x})$ assuming 3 alternative training sample points, respectively. If the starting point is at $\mathbf{x}_c$, the local search strategy now converges to a local optimum that provides significant fitness improvement over the initial point $\mathbf{x}_c$ with $f(\mathbf{x}_{lo}) < \hat{f}(\mathbf{x}_{lo})$ and $f(\mathbf{x}_{lo}) < f(\mathbf{x}_c)$, thus accelerating evolutionary search. Blessed by the presence of uncertainty or approximation errors in the surrogate, SAMA converges to the global optimum of the original fitness function in a fast track mode, i.e., $\mathbf{x}_{lo}$ corresponds to the global optimum of the original fitness function.

5.1.2 Empirical Study on Impact of Uncertainty

To demonstrate the impact of uncertainty on SAMA search, an empirical study in this subsection is presented for some commonly used benchmark test problems with diverse fitness landscapes. Among them are unimodal Sphere and Step functions and three highly multimodal benchmark problems that include Ackley, Griewank and Rastrigin functions. The global optimum (i.e., minimum) solution of all 20-dimensional benchmark problems considered is located at zero. Greater details on the benchmark problems used are provided in the Appendix A. Besides the standard GA, standard MA, SAMA-PR and SAMA-RBF, the results of SAMA-Perfect is also reported for comparisons.

An important question that arises when choosing approximation methods in practice is whether one should use interpolation or regression techniques to construct surrogates when the observed data is generated using computer models. Here, since the primarily

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

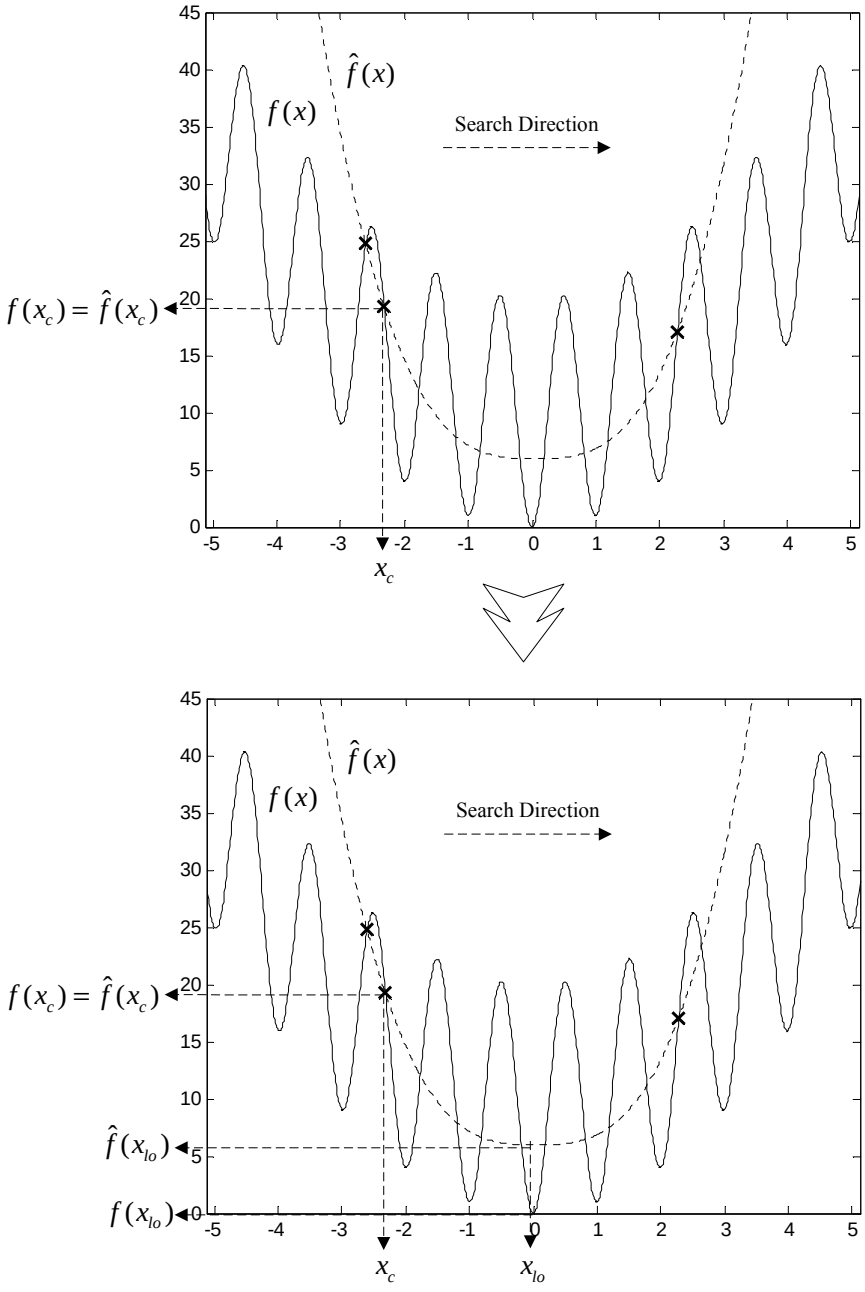


Figure 5.2: 'Blessing of uncertainty' in surrogate-assisted memetic algorithm

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

concern with approximating deterministic computer models is assumed to not suffer from numerically induced convergence or discretization noise, perfectly interpolating models are most germane to my concerns. The effect of '*curse of uncertainty*' on evolutionary search is thus decreased for the training samples when using exact interpolating models. In contrast, regressing techniques do not model the training data exactly but generalizes from the training samples, for example, in a least square sense. To investigate and utilize the effects of '*curse and blessing of uncertainty*', the least-squares regression based on low-order polynomial regression (PR) model is used such that the approximation can be significantly improved even on the training samples.

In the present study, both PR and RBF models are considered in SAMA, which are referred here as SAMA-PR and SAMA-RBF, respectively. Besides the standard GA, standard MA, SAMA-PR and SAMA-RBF, the results of SAMA-Perfect is also reported for comparisons. SAMA-Perfect refers to an SAMA that employs a conceptual approximation technique that is assumed to generate error-free surrogates, i.e., $rmse = 0$. Hence the notion of '*curse or blessing of uncertainty*' do not exists in a SAMA-Perfect search. As such, any SAMA that perform worse or better than the SAMA-Perfect is clearly attributed to the effects of '*curse and blessing of uncertainty*', respectively. Note the difference between MA and SAMA-Perfect. In SAMA-Perfect, the exact fitness evaluations generated by the application of local search are not considered as part of the total exhausted exact evaluation counts in the stopping criteria.

5.1.2.1 Impact of Uncertainty on SAMA Variants

In present empirical study, a standard GA with population size of 50, 1-point crossover and mutation operators at probabilities 0.6 and 0.01, respectively, is employed. However, apart from the standard GA settings, one user-specified parameter of the SAMA is the number of training data points used to construct the surrogate, M. Since $(d+1)(d+2)/2$

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

is the minimum number of data points required for fitting a $2nd$ order PR model, M is set to $(d+1)(d+2)/2$, where d denotes the number of dimension. In the SAMA-Perfect, error-free surrogates are simulated using the exact fitness function. Further, online build surrogate model is considered by using localized training data, i.e., the M nearest neighbors of an individual are selected from the archived database containing previously searched data points. Note that the database is continuously updated as the search proceeds. The feasible sequential quadratic programming (FSQP) is then used as the local search solver in both the MA and SAMA variants.

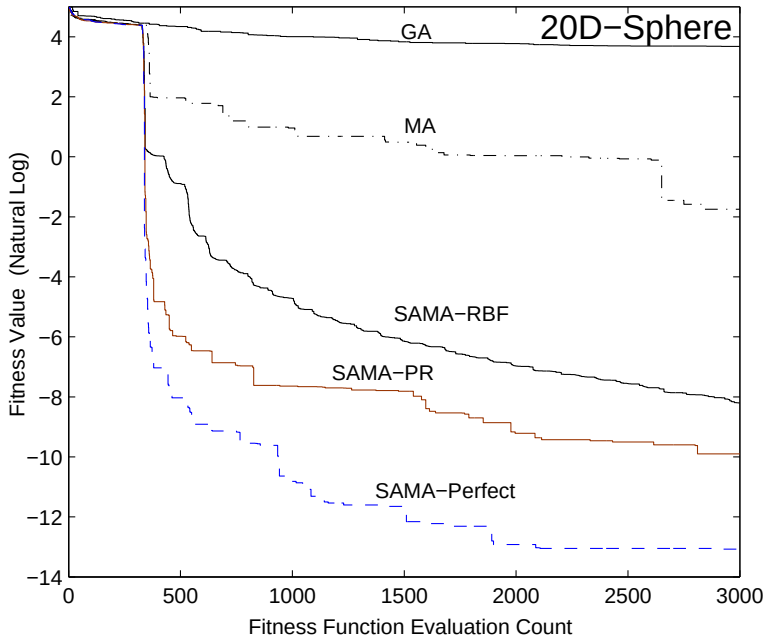


Figure 5.3: Convergence trends of the GA, SAMA-PR, SAMA-RBF and SAMA-Perfect on 20D Sphere function

The convergence trends obtained for the standard GA, standard MA, SAMA-PR, SAMA-RBF and SAMA-Perfect on the five benchmark problems are plotted in Figs. 5.3-5.7. The results presented are averaged over 20 independent runs conducted with a tractable computational budget of 3×10^3 exact fitness function evaluations. The

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

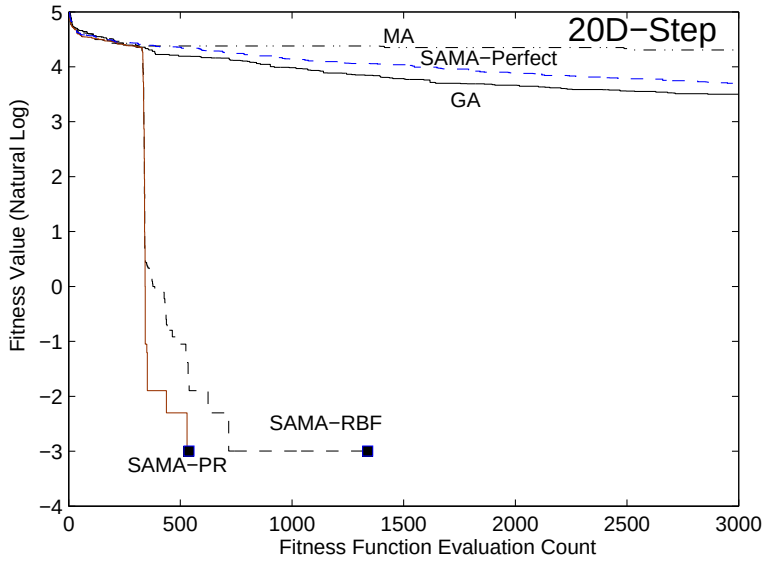


Figure 5.4: Convergence trends of the GA, MA, SAMA-PR, SAMA-RBF and SAMA-Perfect on 20D Step function

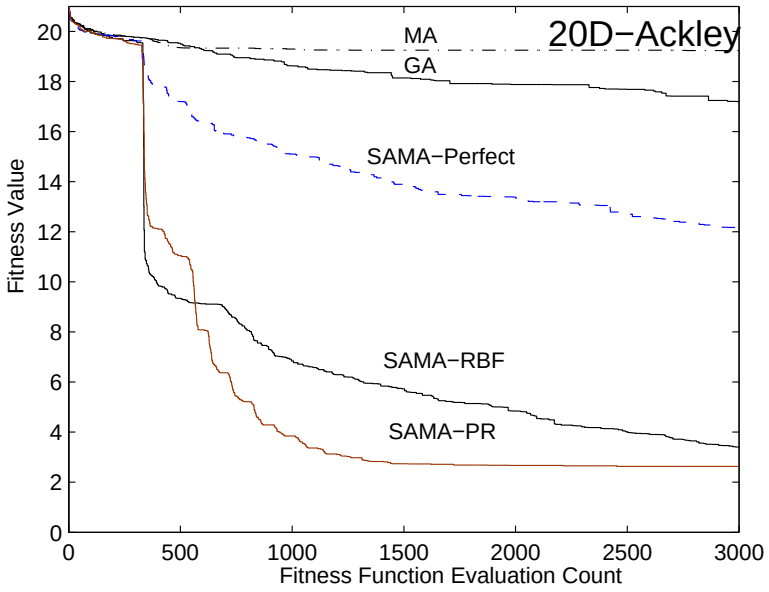


Figure 5.5: Convergence trends of the GA, MA, SAMA-PR, SAMA-RBF and SAMA-Perfect on 20D Ackley function

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

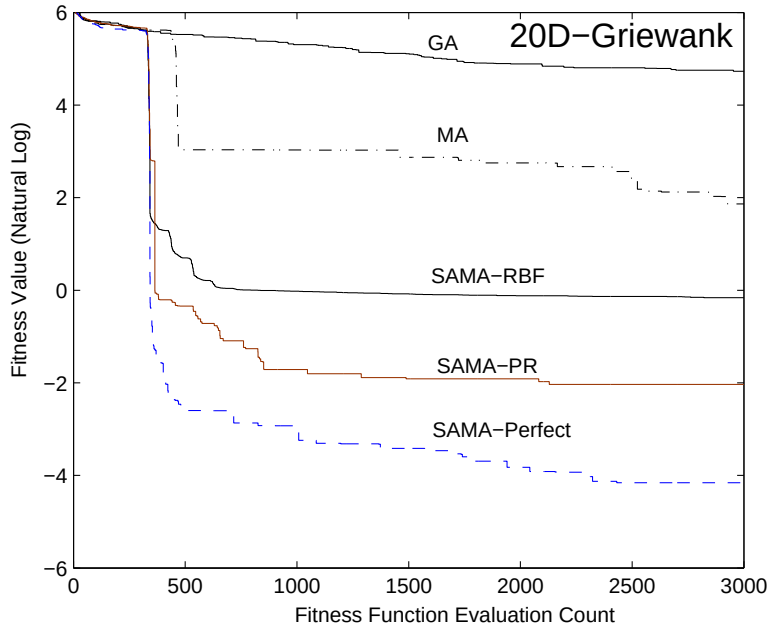


Figure 5.6: Convergence trends of the GA, MA, SAMA-PR, SAMA-RBF and SAMA-Perfect on 20D Griewank function

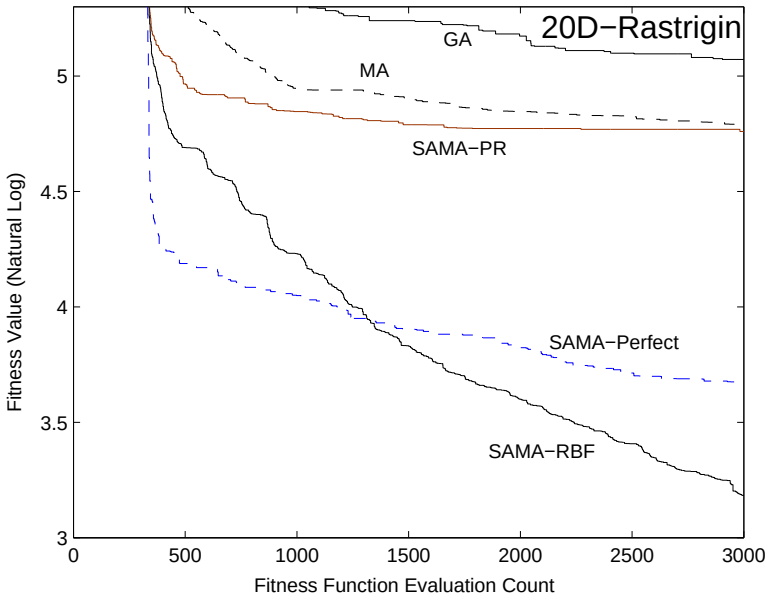


Figure 5.7: Convergence trends of the GA, MA, SAMA-PR, SAMA-RBF and SAMA-Perfect on 20D Rastrigin function

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

black square in the figures denotes a convergence to the true global optimum within the tractable computational budget imposed.

From these results, it is notable that all the SAMA variants considered here, i.e., SAMA-PR, SAMA-RBF and SAMA-Perfect, are capable of converging to good solution quality more efficiently than both standard GA and MA on the benchmark problems. This makes good sense since memetic algorithms, i.e., EAs that employ local search heavily such as SAMA-RBF, generally search more efficiently. Most importantly, since the surrogates are used in place of the exact fitness function when conducting local searches, the SAMA-PR, SAMA-RBF and SAMA-Perfect incurs significantly lower computational efforts than the standard MA for the same search generations.

Further, it is also worth highlighting that both SAMA-RBF and SAMA-PR outperforms the SAMA-Perfect significantly on the Step unimodal function (see Fig. 5.4) and the Ackley multimodal function (see Fig. 5.5). On the Rastrigin function in Fig. 5.7, SAMA-RBF converges to good quality solutions significantly faster than SAMA-Perfect. Clearly, these cases demonstrate the effect of '*blessing of uncertainty*' on multimodal functions in accelerating evolutionary search. The uncertainty in the surrogate model helps smooth the multimodal landscape of the original fitness function, leading to significant speedup in search convergence.

On the other hand, SAMA-Perfect performs better than both SAMA-PR and SAMA-RBF on the Sphere unimodal function and Griewank multimodal function as shown in Fig. 5.3 and 5.6, respectively. It also outperforms SAMA-PR on the Rastrigin multimodal function in Fig. 5.7. Note that these observations demonstrate there exists the effect of *'curse of uncertainty'* due to the negative impact of approximation errors in the surrogate model which could result the evolutionary search in converging at the false global optima.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Benchmark Problems	r correlation		$rmse$	
	PR	**RBF**	**PR**	**RBF**
Ackley	0.9484	0.9979	2.2303	0.3276
Griewank	1.0000	0.9982	0.0154	1.8852
Rastrigin	0.5120	0.5988	98.8377	61.4169
Sphere	1.0000	0.8403	3.5827e-4	0.7937
Step	0.8423	0.9178	1.4635	0.9182

Table 5.1: The $rmse$ and r correlation measures for PR and RBF surrogates

5.1.2.2 Surrogate Quality/Uncertainty and SAMA Search Performance Relations

Next, any forms of relations between the quality of surrogate model or uncertainty to SAMA search performance are investigated.

The quality of a surrogate can be assessed a posteriori by comparing an independent set of exact fitness values with the corresponding fitness estimated using the surrogate. The root mean square error (denoted by $rmse$) and correlation coefficient (denoted by r) [99] are two commonly used measures for evaluating the accuracy of a surrogate.

If $f(\mathbf{x})$ denotes the original fitness function and the approximated function is $\hat{f}(\mathbf{x})$, the approximation error at any design point $\mathbf{x}_i$ is $e(\mathbf{x}_i)$, i.e., the uncertainty introduced by the surrogate at $\mathbf{x}_i$, may then be defined as

$$e(\mathbf{x}_i) = |f(\mathbf{x}_i) - \hat{f}(\mathbf{x}_i)| \tag{5.5}$$

For each non-duplicated individual, if n fitness calls to $\hat{f}(\mathbf{x})$ are made in the SAMA local search strategy, the root mean square error, $rmse$ can be derived as

$$rmse = \sqrt{\frac{\sum_{i=1}^{n} e^2(\mathbf{x}_i)}{n}}. \tag{5.6}$$

The correlation coefficient r is defined by

$$r = \frac{N\sum t\hat{t} - \sum t \sum \hat{t}}{\sqrt{[N\sum t^2 - (\sum t)^2][N\sum \hat{t}^2 - (\sum \hat{t})^2]}}. \tag{5.7}$$

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

where N is the sample size used. t denotes the exact fitness values from the independent test set, and $\hat{t}$ are the fitness values estimated by the surrogate. If a correlation coefficient of 1 is obtained, this implies that the surrogate models the test set accurately.

Here, locally optimized solutions using PR or RBF models in the SAMA run are archived to form the sample set for evaluating the quality of different surrogates. The obtained $rmse$ and r measures when using PR or RBF models on the benchmark problems are reported in Table 5.1. The $t - \hat{t}$ correlation plots of the PR and RBF models on the 20D Ackley function are also depicted in Figures 5.8 and 5.9. Similar plots for the 20D Rastrigin function are provided in Figures. 5.10 and 5.11.

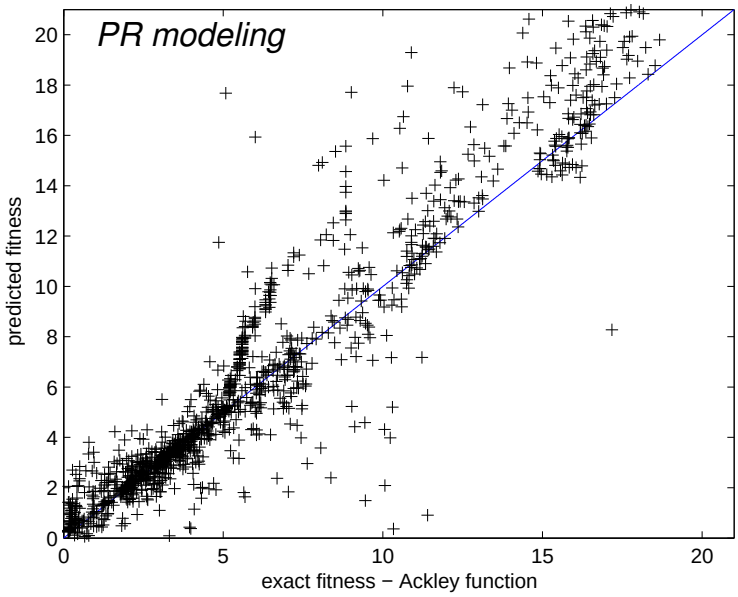


Figure 5.8: $t - \hat{t}$ plot of PR modeling for 20D Ackley function

The results obtained indicate that linear-Spline RBF outperforms the 2nd order PR on 3 out of the 5 benchmark problems, in terms of approximation accuracies $rmse$ or r but under performs on the other 2 problems considered, see Table 5.1. Although this leads to better search performances on the Rastrigin function in SAMA-RBF than SAMA-PR (see Figure 5.7), the same conclusion cannot be drawn on Ackley, see Figure 5.5. This

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

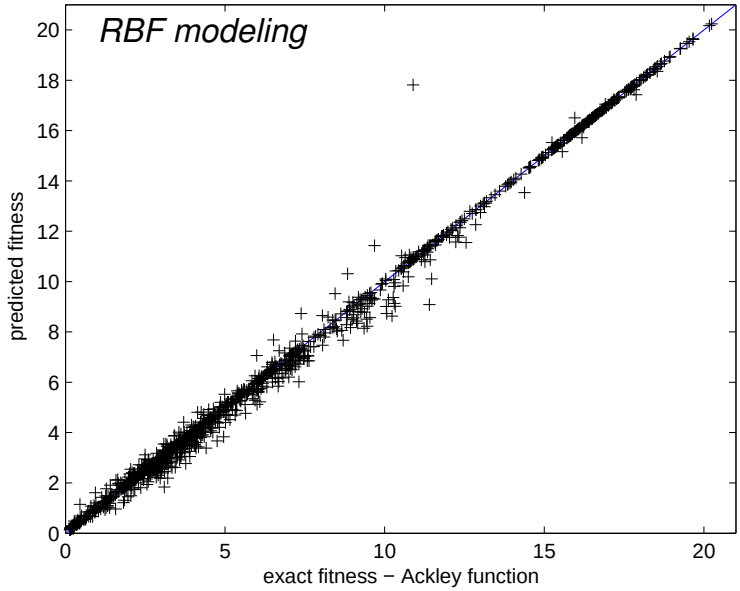


Figure 5.9: $t - \hat{t}$ plot of RBF modeling for 20D Ackley function

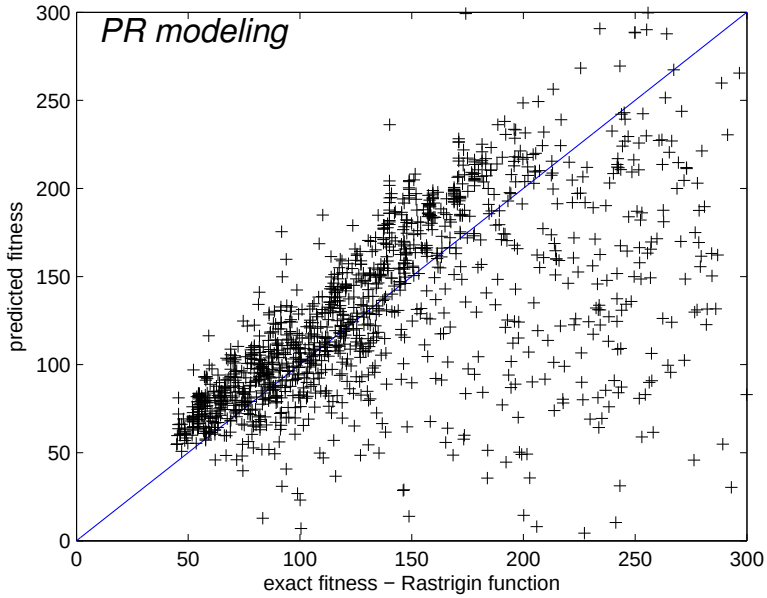


Figure 5.10: $t - \hat{t}$ plot of PR modeling for 20D Rastrigin function

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

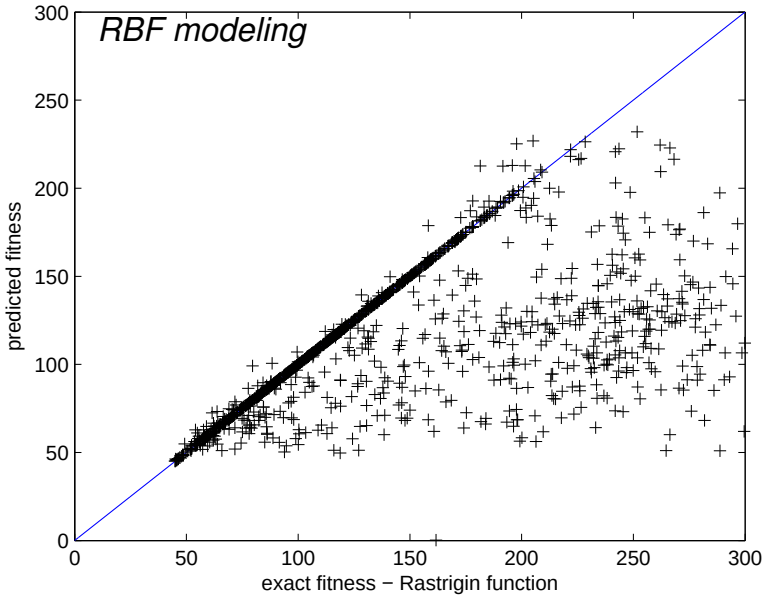


Figure 5.11: $t - \hat{t}$ plot of RBF modeling for 20D Rastrigin function

implies that there remains to be no clear relationship between accuracy of the surrogate models or the effect of '*cursing and blessing of uncertainty*' to SAMA search performance that one could leverage for appropriate model selection.

5.2 Multi-Surrogates Assisted Memetic Algorithm

From the numerical results obtained in subsection 5.1.2, it is worth keeping in mind that uncertainty introduced by approximation errors in the surrogate model can benefit as well as obstruct effective optimization search. Hence, in contrast to mitigating only the effects of '*curse of uncertainty*' by seeking for high accuracy approximation models (like in most existing works [6], [62], [63], [64]), it would be wise to also consider leveraging from possible benefits of '*blessing of uncertainty*' when designing new SAEAs. Since there is often no prior knowledge on which surrogates would be most appropriate for the problem of interest as show in subsection 5.1.2.2, the use of multi-surrogates in the

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

SAMA search, i.e., MSAMA, is proposed to solve computationally expensive optimization problems efficiently.

5.2.1 Outline of MSAMA Algorithm

The basic steps of proposed MSAMA are outlined in Procedure 8 and the work flow of MSAMA is depicted in Figure 5.12. The key difference between MSAMA and SAMA lies in the local search phase. In MSAMA, multi-local searches based on dissimilar surrogates, i.e., $\{\hat{t_1}, \hat{t_2}, ..., \hat{t_j}\}$ are conducted in parallel [100], where j represents the number of surrogates considered.

Note that surrogate ensemble [69] may also be considered as one of independent surrogate model used in MSAMA. Subsequently, the best improved solution obtained in the local learning phase then proceeds with the Lamarckian Learning process. Since the local searches and the construction of local surrogates are conducted in parallel, the computational efforts incurred remains significantly lower than using the exact simulation codes.

Further, by using multiple surrogate models that exhibits different degree of '*curse and blessing of uncertainty*', the proposed MSAMA can accelerate evolutionary search in the following ways:

(i) If all the surrogate models used display effects of '*blessing of uncertainty*', benefit is achieved by leveraging from either of the model's blessing.

(ii) If none of the surrogate models display any effects of '*blessing of uncertainty*', benefit can still be attained by leveraging from the model that provides the minimum prediction errors.

(iii) If some surrogate models display effects of '*blessing of uncertainty*', while others on '*curse of uncertainty*', benefits can still be attained by leveraging from models that provides blessing on the search.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

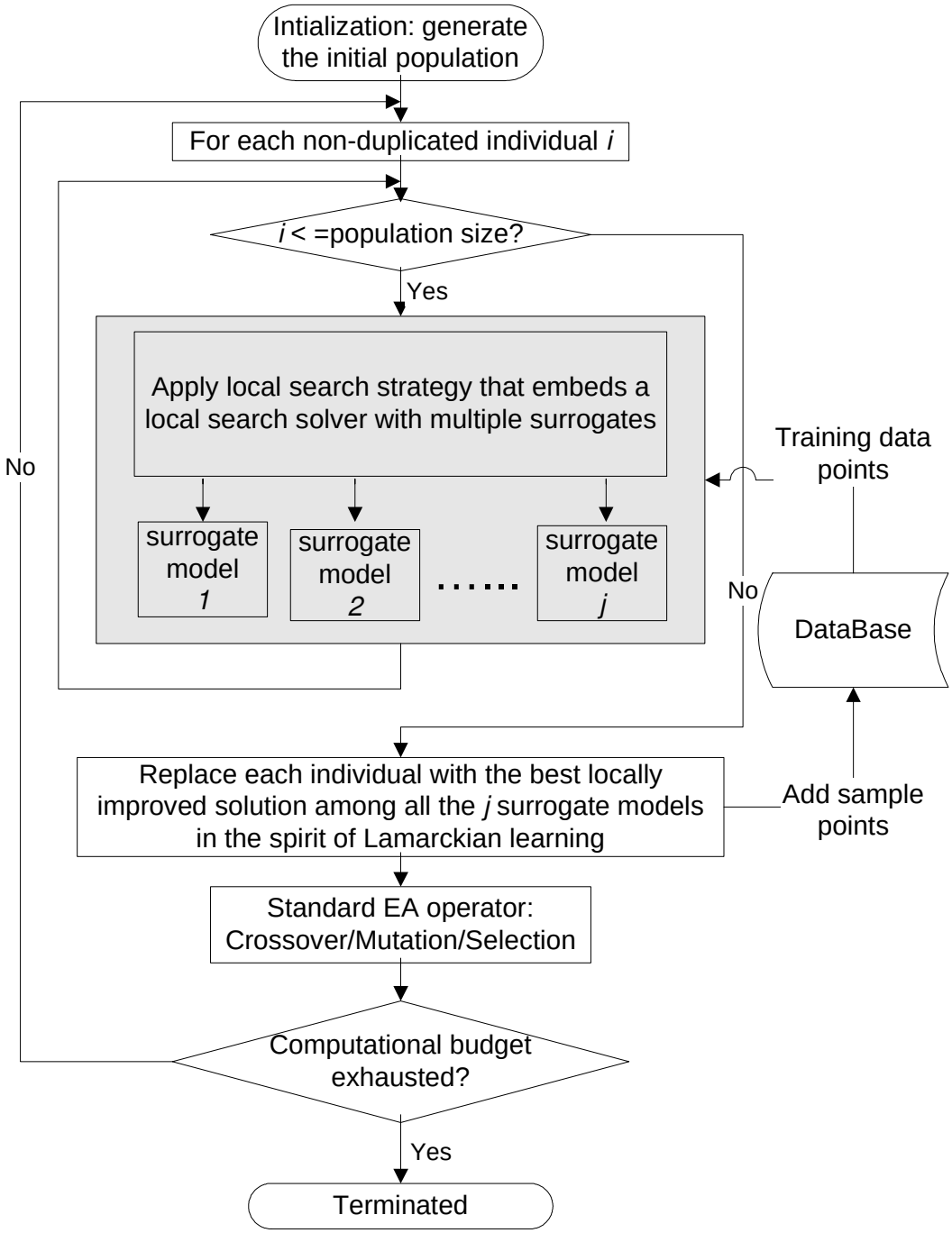


Figure 5.12: Work Flow of MSAMA

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Procedure 8 MSAMA

/*Phase 0: Initialization */
$gen := 0$; /* Initialize the generation counter. */
$pop[gen] := CreateInitPop(db)$; /*Create an initial population from database*/
while (*computational budget is not exhausted*) **do**
$fitness[gen] := Evaluate(pop[gen], f(\mathbf{x}))$; /*Evaluate population using $f(\mathbf{x})$*/

/*Phase 1: Multi-Surrogates Assisted Local Search Strategy */
for (each non-duplicated individual in $pop[gen]$) **do**
for $j = 0$ to $MaxModelNum$ **do**
/*Select dissimilar surrogate modeling techniques for each iteration*/
$SelectModelingMethod(j)$;
/*Construct a online local surrogate model with selected modeling technique*/
$\hat{f}_l^j(\mathbf{x}) := BuildLocalModel(db)$;
/*Apply local searches with $\hat{f}_l^j(\mathbf{x})$*/
$LIpop^j[gen] := LocalImprovement(\hat{f}_l^j(\mathbf{x}))$;
$LIfitness^j[gen] := Evaluate(LIpop[gen], \hat{f}_l^j(\mathbf{x}))$;
/*Keep the best locally improved individuals*/
if $LIfitness^j[gen]$ is the best **then**
$BESTpop[gen] \Leftarrow BESTpop^j[gen]$;
end if
end for
/*Evaluate the best local improvements*/
$BESTfitness[gen] := Evaluate(BESTpop[gen], f(\mathbf{x}))$;
/*Replace the individual with locally improved solution.*/
$pop[gen] \Leftarrow BESTpop[gen]$;
$fitness[gen] \Leftarrow BESTfitness[gen]$;
end for

/*Phase 2: Standard EA Operations */
for $i = 0$ to $MaxPopSize$ **do**
$(parent1[gen], parent2[gen]) := Select(pop[gen])$;
$child[gen] := Crossover(parent1[gen], parent2[gen])$;
$Mutation(child[gen])$;
end for
$fitness[gen] := Evaluate(pop[gen], f(\mathbf{x}))$; /*Evaluate population using $f(\mathbf{x})$*/
$pop[gen+1] := CreateNewPop(child[gen])$.
$gen := gen + 1$; /* Increase the generation counter. */
end while

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

PARAMETER	VALUE
population size	50
crossover probability	0.6
mutation rate	0.01
number of nearest neighboring data points M	$(d+1)(d+2)/2$

Table 5.2: The configurations of SAMA and MSAMA used in empirical study.

5.2.2 Empirical Study on MSAMA

To investigate the efficacy of the proposed MSAMA, an empirical study is conducted on previously considered five benchmark problems having diverse fitness landscapes. The configurations of all parameters in the algorithms are kept consistent to the previous study of SAMAs in Table 5.2.

When it comes to model selection in the MSAMA, it is obvious to favor surrogates that provide maximum search improvements. A clear choice of surrogate in the MSAMA is one with perfect prediction accuracy, i.e. perfect model, capable of resolving the '*curse of uncertainty*'. On the other hand, to benefit from the effects of '*blessing of uncertainty*', any surrogates that is capable of generalizing the multi-modality properties of the problem landscape or providing good search directions should be considered. To facilitate a diverse pool of surrogate models in the MSAMA, both exact interpolating and regression approximation techniques are used since they produce fundamentally very different surrogates.

Three types of surrogate models are used in MSAMA in present empirical study. These include the linear-spline exact interpolating RBF, 2nd order regression PR, and a conceptual approximation method that generates error-free surrogates, i.e., a perfect model. The MSAMA approach uses two out of three types of surrogate models. The abbreviations of the MSAMA variants are tabulated in Table 5.3.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Abbreviation	SAMA or MSAMA Variants
SAMA-PR	SAMA with PR model
SAMA-RBF	SAMA with RBF model
SAMA-Perfect	SAMA with Perfect model
MSAMA-EP	MSAMA with Perfect and PR models
MSAMA-EF	MSAMA with Perfect and RBF models
MSAMA-PF	MSAMA with PR and RBF models

Table 5.3: Abbreviation of SAMA and MSAMA Variants

5.2.2.1 MSAMA using Perfect Model

Next, the performance of MSAMA is investigated by using the perfect model and either PR regression or RBF interpolation model. Figures 5.13-5.17 depict the search trends of MSAMA-EP and MSAMA-EF in comparison to the SAMA variants (i.e., SAMA-PR, SAMA-RBF and SAMA-Perfect) on all the 20 dimensional benchmark problems. It is noted that the search trends of SAMA-Perfect is omitted from Figs. 5.14 and 5.15 for comparison due to their significantly poor results on the Step and Ackely functions. The black square in the figures denotes that convergence to the true global optimum has reached within the tractable computational budget imposed.

The results obtained on the unimodal benchmark problems are discussed firstly. Sphere is a nonlinear, continuous, convex smooth function. Since the landscape of the Sphere function is quadratic in nature, a 2nd order PR model clearly serves as more appropriate than the linear-spline RBF. This explains the improved search performance by SAMA-PR over SAMA-RBF on the Sphere function. Nevertheless, neither SAMA-PR nor SAMA-RBF could outperform the SAMA-Perfect which demonstrates the presence of '*curse of uncertainty*'. Further, both MSAMA-EP and MSAMA-EF can achieve better solution than all the SAMA counterparts. For instance, MSAMA-EF converges to the global optimum in less than 1.5×10^3 exact fitness function evaluations. The perfect model used in MSAMA-EF and MSAMA-EP help resolve any forms of '*curse of uncertainty*' that exists. At the same time, MSAMA benefits from the effects of '*blessing of*

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

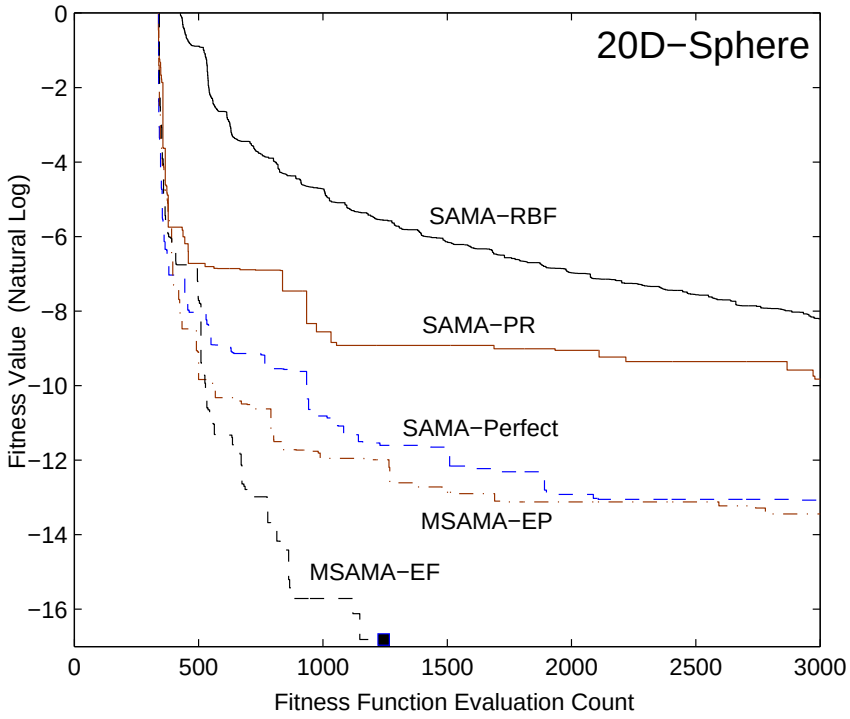


Figure 5.13: Convergence trends of the MSAMA variants on 20D Sphere function

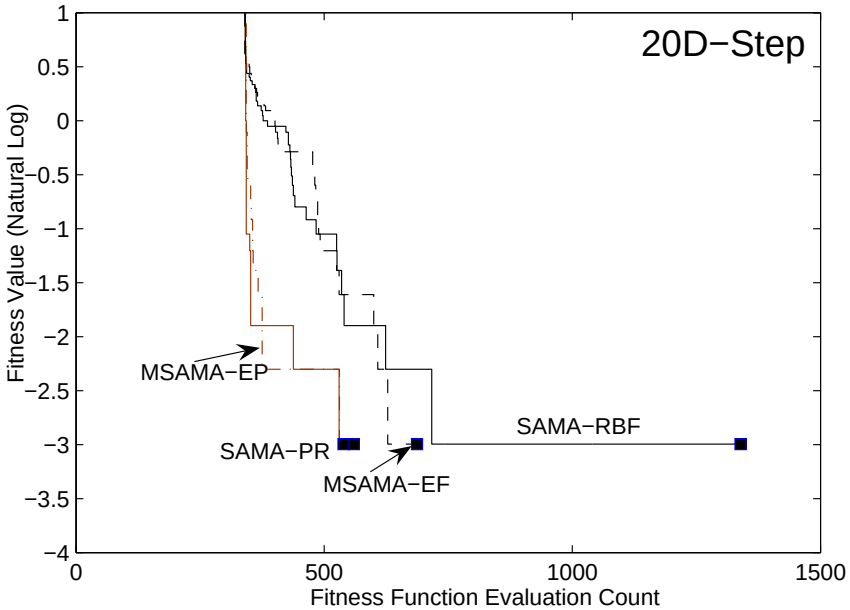


Figure 5.14: Convergence trends of the MSAMA variants on 20D Step function

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

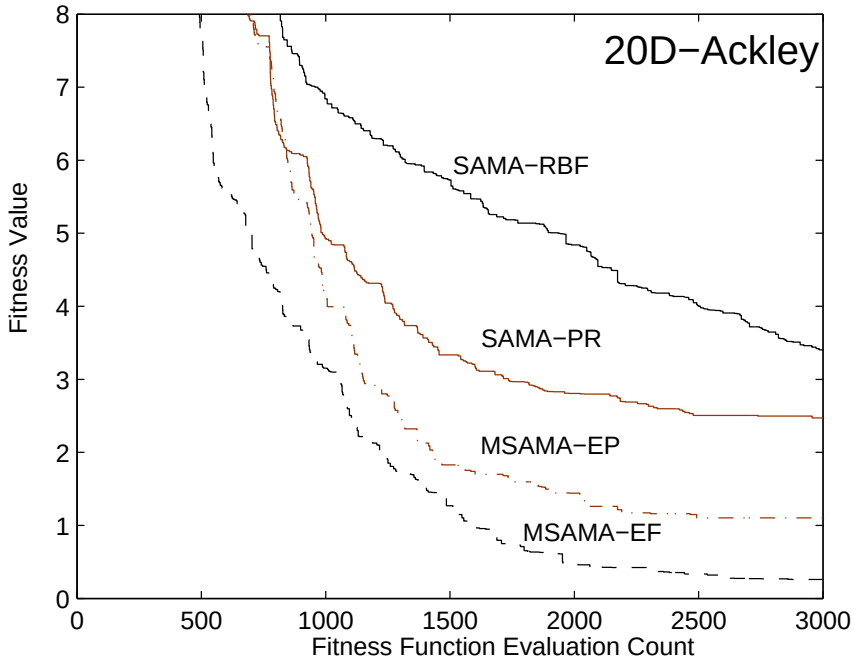


Figure 5.15: Convergence trends of the MSAMA variants on 20D Ackley function

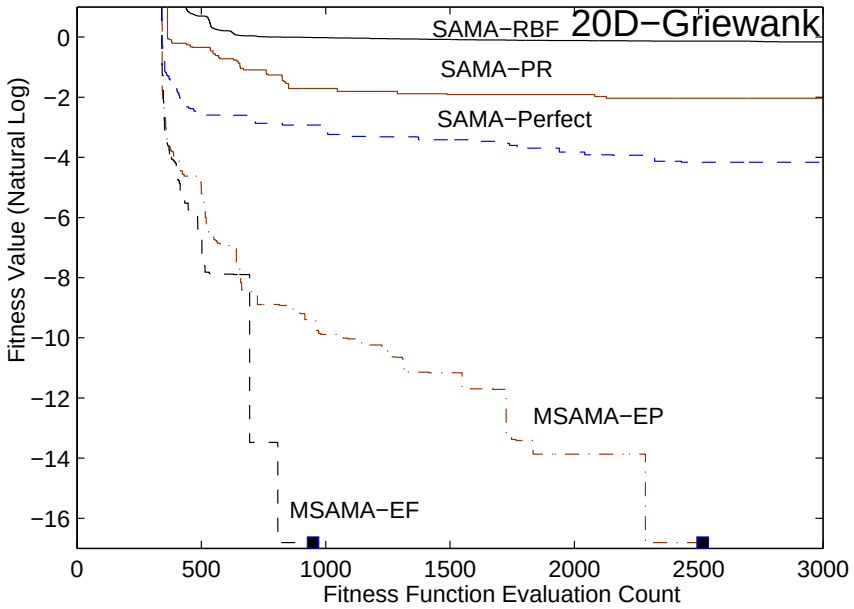


Figure 5.16: Convergence trends of the MSAMA variants on 20D Griewank function

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

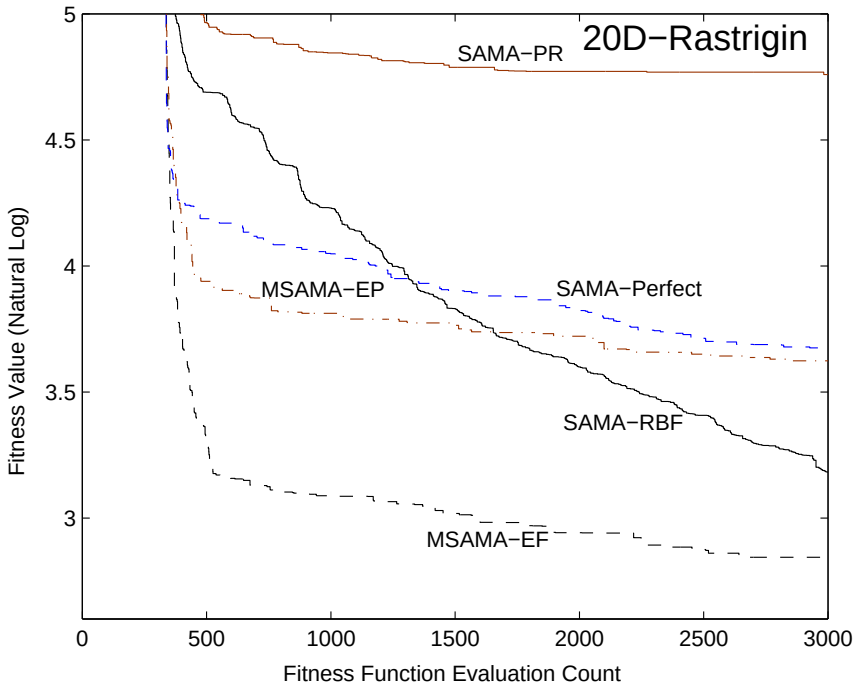


Figure 5.17: Convergence trends of the MSAMA variants on 20D Rastrigin function

uncertainty' introduced by either PR or RBF approximations, leading to improved search performances.

The Step function consist of flat plateaus with $slope = 0$ in an underlying continuous function. In contrast to Sphere, it is hard for any gradient-based optimization algorithm to locate the global optimum of the Step function since minor changes of the problem variables do not affect the fitness value. However, the surrogate modeling techniques in SAMA helps smooth the discontinuities of the step landscape,i.e., an effect of '*blessing of uncertainty*' on SAMA search, thus leading to the superior search performances displayed by both SAMA-PR and SAMA-RBF (i.e., the global optimum is found in less than 1.5×10^3 exact fitness function calls). Once again, both MSAMA-EP and MSAMA-EF converges to the global optimum more efficiently than all the SAMA counterparts.

Consider next the multimodal benchmark problems. The results in Figures 5.15-5.17

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

also indicate that both MSAMAs, i.e., MSAMA-EP and MSAMA-EF, converges to the global optimum of the benchmark problems or improved quality solutions significantly faster than the SAMA counterparts, once again demonstrating the positive benefits of '*blessing of uncertainty*' introduced by either PR or RBF approximation on MSAMA search.

5.2.2.2 MSAMA using both Regression and Interpolation Models

Since an approximation technique that produces perfect model of zero *rmse* does not exists in reality, the exact interpolating RBF models are considered as a possible substitute in practice since its *rmse* is zero on the training samples. The 2nd order PR on the other hand fits in nature as an appropriate least-squares regression model that possess abilities for generalizing the landscape multi-modality. Since PR regression model provides fundamentally different approximation compared to linear-spline RBF exact interpolation, it will ensure the diversity of surrogate models used in MSAMA. The MSAMA that employs both PR and RBF surrogates in the local search phase is referred as MSAMA-PF.

The search traces obtained by MSAMA-PF, SAMA-PR and SAMA-RBF for solving the benchmark problems are plotted in Figs. 5.18-5.22. These results again highlights the superior performance of the MSAMA over the SAMA counterparts where only a single approximation method, i.e., either PR or RBF model, is considered. In particular, the MSAMA-PF converges to the global optimum in less than 4×10^2 exact fitness evaluations on the Step function. It also converges to better solution within the computational budget imposed, on all the other problems considered.

Further statistical significance in the search performances of MSAMA-PF, SAMA-PR and SAMA-RBF can be arrived from the box plots provided in Figs. 5.23-5.26 which shows the median and the variance of the fitness value for the best found individual across 20 independent runs. No box plot for the Step function is provided since both SAMAs

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

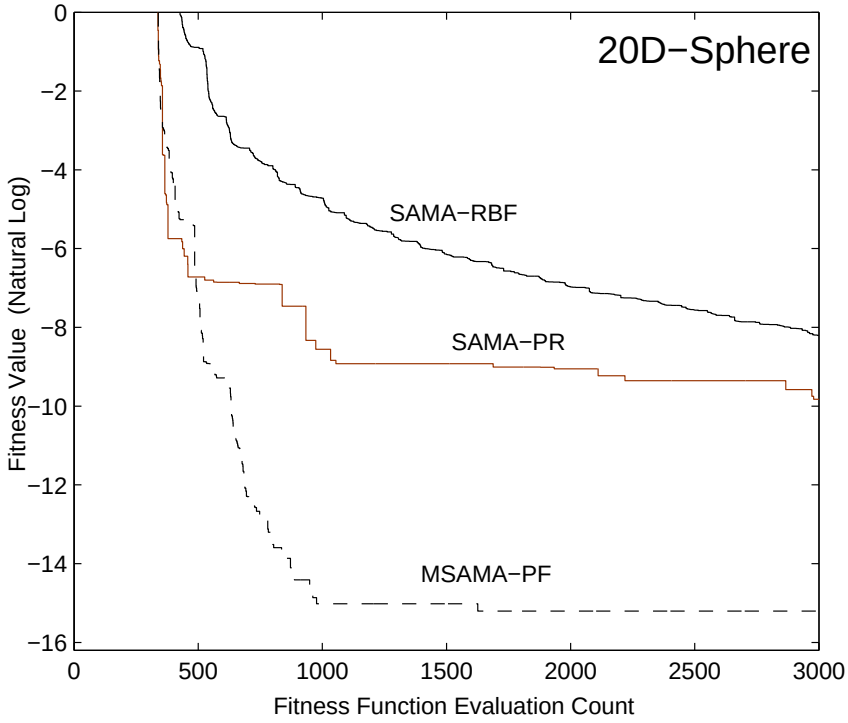


Figure 5.18: Convergence trends of the MSAMA-PF on 20D Sphere function

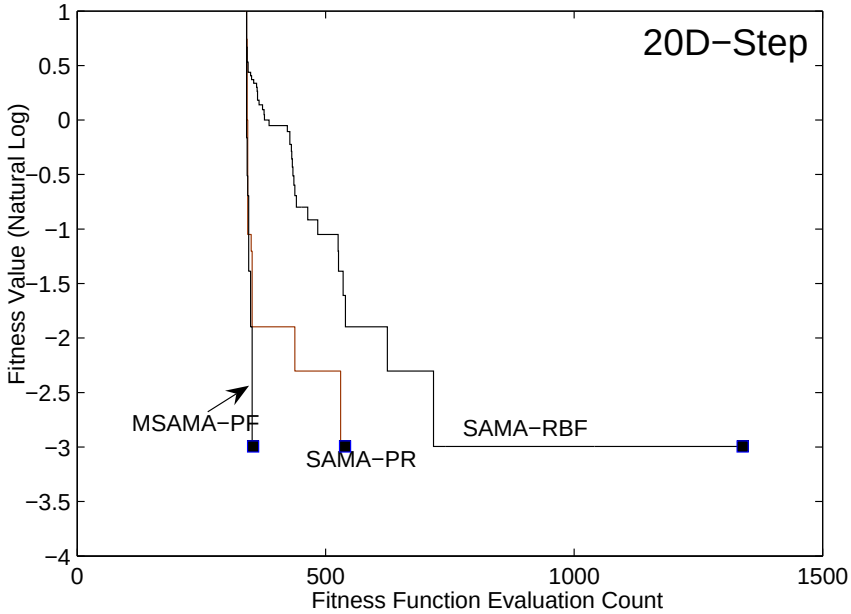


Figure 5.19: Convergence trends of the MSAMA-PF on 20D Step function

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

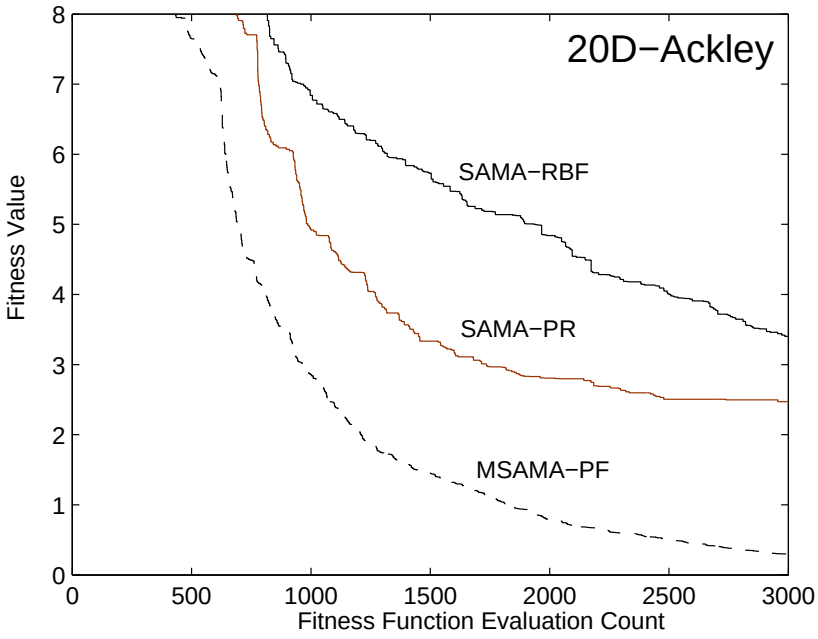


Figure 5.20: Convergence trends of the MSAMA-PF on 20D Ackley function

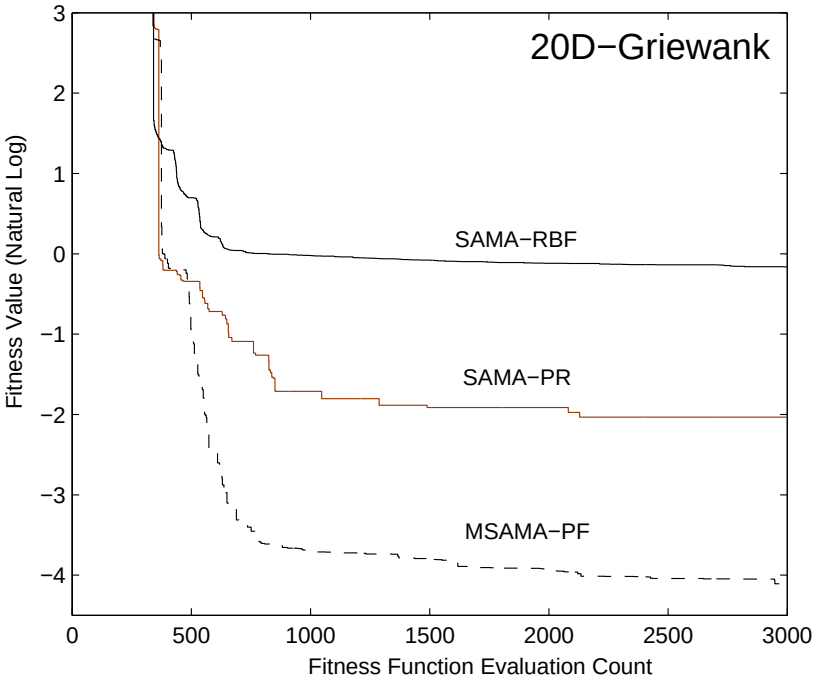


Figure 5.21: Convergence trends of the MSAMA-PF on 20D Griewank function

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

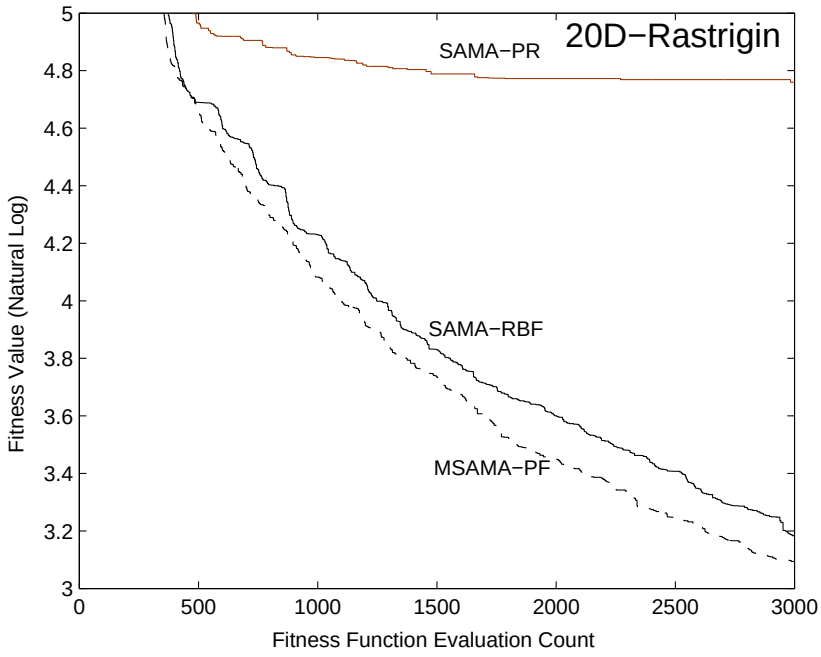


Figure 5.22: Convergence trends of the MSAMA-PF on 20D Rastrigin function

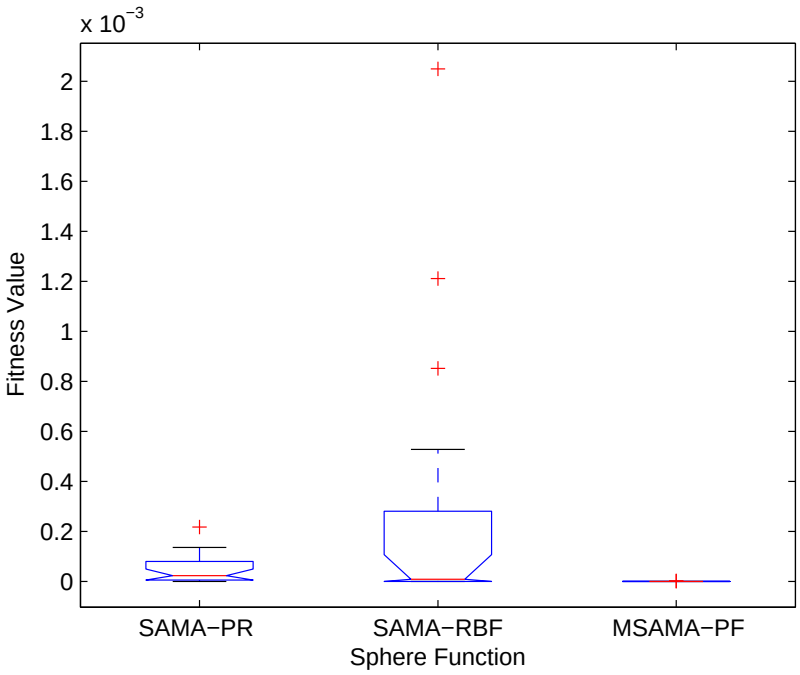


Figure 5.23: SAMA-PR, SAMA-RBF and MSAMA-PF Box plot of the best fitness in the final generation for 20D Sphere function

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

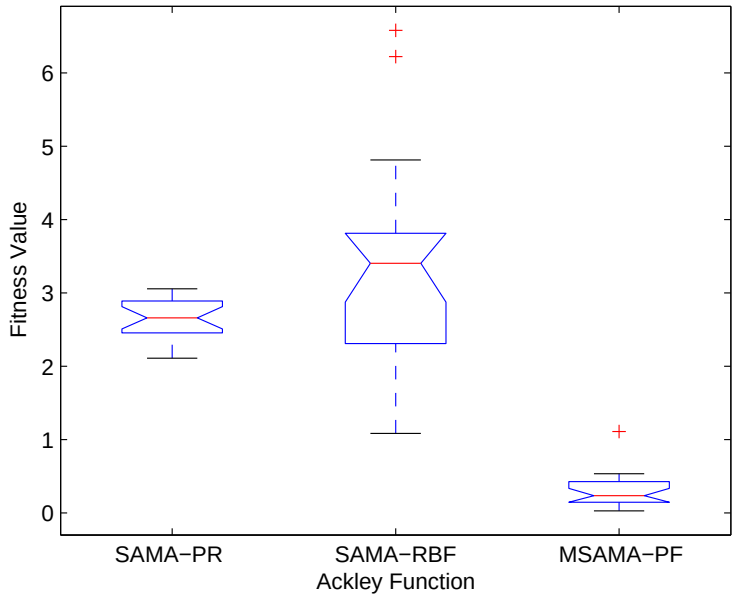


Figure 5.24: SAMA-PR, SAMA-RBF and MSAMA-PF Box plot of the best fitness in the final generation for 20D Ackley function

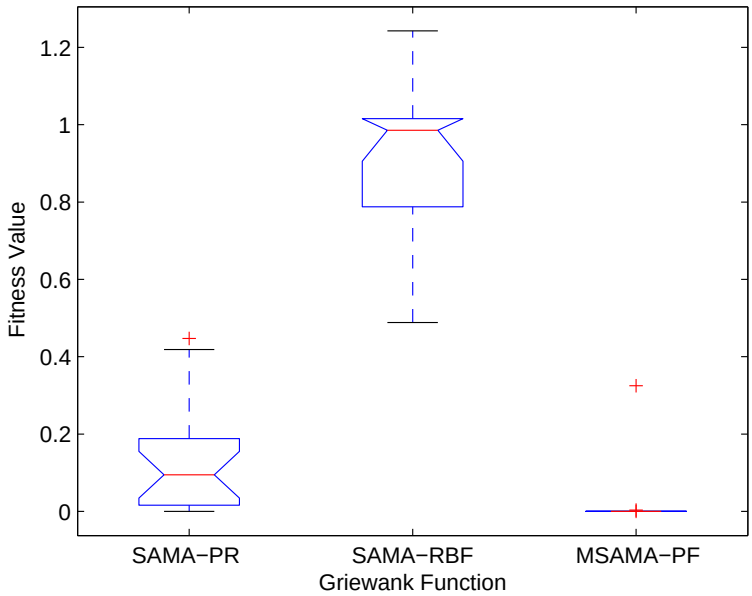


Figure 5.25: SAMA-PR, SAMA-RBF and MSAMA-PF Box plot of the best fitness in the final generation for 20D Griewank function

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

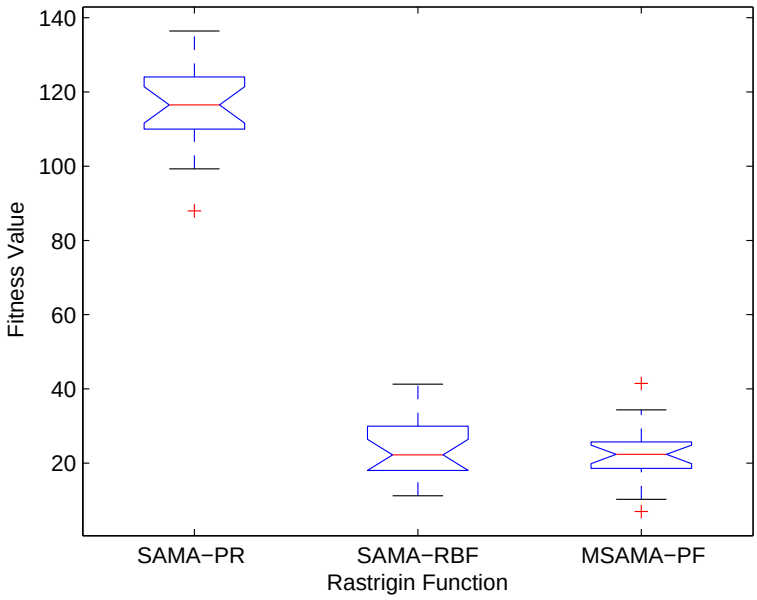


Figure 5.26: SAMA-PR, SAMA-RBF and MSAMA-PF Box plot of the best fitness in the final generation for 20D Rastrigin function

CRITERIA	DEFINITION
Best	The best fitness value among all independent runs
Worst	The worst fitness value among all independent runs
Mean	The mean/average fitness value for all independent runs
Median	The median fitness value among all independent runs
Deviation	The deviation value for all independent runs

Table 5.4: Criteria for measuring performance.

and MSAMA-PF converge to the global minimum within the imposed computational budget (see Fig. 5.19). In a notched box plot, the notch represents a robust estimate of the uncertainty about the median. The lower and upper bounds of the box are the 25% and 75% lower quartiles. The whiskers are lines extending from each end of the box to show the extent of the rest of the data. Outliers are data with values beyond the ends of the whiskers and denoted by the '+' sign. Boxes whose notches do not overlap indicate that the medians of the two groups differ at the 5% significance level.

On the unimodal Sphere function (Fig. 5.23), SAMA-PR, SAMA-RBF as well as

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Algorithm Type	Performance Criteria				
	Best	Worst	Mean	Median	Deviation
SAMA-PR	0	0.2180	0.0478	0.0210	0.0575
SAMA-RBF	0	0.0021	2.7e-4	0	5.4299e-4
MSAMA-PF	0	3.0e-6	2.5e-7	0	7.1635e-7

Table 5.5: Statistics of SAMA-PR, SAMA-RBF and MSAMA-PF for Sphere Function.

Algorithm Type	Performance Criteria				
	Best	Worst	Mean	Median	Deviation
SAMA-PR	2.1086	3.0550	2.6352	2.6601	0.2785
SAMA-RBF	1.0831	6.5819	3.3487	3.4032	1.4313
MSAMA-PF	0.0261	1.1101	0.298	0.234	0.2467

Table 5.6: Statistics of SAMA-PR, SAMA-RBF and MSAMA-PF for Ackley Function.

Algorithm Type	Performance Criteria				
	Best	Worst	Mean	Median	Deviation
SAMA-PR	0	0.4475	0.1308	0.0949	0.1354
SAMA-RBF	0.4883	1.2427	0.9017	0.9854	0.1869
MSAMA-PF	0	0.3249	0.0164	0	0.0726

Table 5.7: Statistics of SAMA-PR, SAMA-RBF and MSAMA-PF for Griewank Function.

Algorithm Type	Performance Criteria				
	Best	Worst	Mean	Median	Deviation
SAMA-PR	87.9489	136.4428	116.7137	116.5195	11.4290
SAMA-RBF	11.2311	41.2498	24.1048	22.2128	8.5925
MSAMA-PF	6.9430	41.4958	22.0591	22.3556	8.2804

Table 5.8: Statistics of SAMA-PR, SAMA-RBF and MSAMA-PF for Rastrigin Function.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

MSAMA-PF all arrive at very good solutions with overlapping notches. Nevertheless, only MSAMA-PF arrives at the global optimum in all the 20 independent runs on the Sphere function. The results in Figs. 5.24-5.25 also indicate that MSAMA-PF outperforms the two SAMA counterparts significantly on Ackley and Griewank functions. There is however no significant difference between SAMA-RBF and MSAMA-PF on the Rastrigin function. For a detailed comparisons between SAMA-PR, SAMA-RBF and MSAMA-PF, the reader is referred to more statistics provided in Tables 5.4-5.8.

5.3 Summary

In this chapter, the negative and positive impacts of the uncertainty introduced by approximation errors in the surrogate on evolutionary optimization, otherwise referred to as '*curse and blessing of uncertainty*', have been introduced and demonstrated.

Empirical studies are presented for a number of unimodal and multimodal benchmark test functions to illustrate the impact of uncertainty using the standard SAMA. The experimental results are compared with those obtained using a standard GA, MA and SAMA with PR, RBF or a perfect surrogate. The results obtained demonstrate the effect of '*curse of uncertainty*' and '*blessing of uncertainty*' attributed by approximation errors in the surrogate-assisted evolutionary algorithms.

Taking this cue, the MSAMA framework is proposed for enhancing evolutionary search performance by leveraging the benefits of '*curse and blessing of uncertainty*' among different surrogate models. Empirical results obtained on both unimodal and multimodal problems further demonstrate that MSAMA outperforms the standard GA, standard MA and other SAMAs significantly, thus making it a promising approach for solving computationally expensive optimization problems.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Chapter 6

Conclusions and Future Work

6.1 Conclusions

The central theme of this thesis is on using multiple surrogate modeling techniques in memetic algorithm for solving costly CED optimization problem under a tractable computational budget. The main contributions of the dissertation work are summarized as follows:

Firstly, the effects of the genetic operators on two SAMA variants, i.e., the standard SAMA and SAMA-TRF, are investigated and compared. Empirical result shows that the genetic operators have different effects on the search performance of standard SAMA and SAMA-TRF. The SAMA-TRF is shown to be more robust against the changes in genetic operators than the standard SAMA.

Next, motivated by the lack of suitable multi-layer SAMA for solving computationally expensive optimization problem, a novel HSAMA which combines both global and local surrogate models is proposed and investigated. Empirical study on the proposed HSAMA optimization framework shows that it leads to great saving in term of computational cost and converges to better solution quality when compared to the standard GA, SAMA-TRF and other existing SAEA optimization frameworks.

Finally, inspired by the notion of '*curse and blessing of uncertainty*' in approximation models, a novel MSAMA for solving optimization problems with computationally

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

expensive fitness evaluation is also presented. Both regression and exact interpolating surrogate modeling techniques are used in the empirical study on memetic search. The results are presented for a series of commonly used benchmark problems to demonstrate that the proposed framework converges to good solution quality more efficiently than the standard GA, MA and other SAMA variants with only one surrogate modeling technique when solving computationally expensive optimization problems. This leads to a new paradigm in SAEA design.

6.2 Future Work

In this thesis, two novel SAMAs that employ multiple surrogate models have been investigated and reported. A summary of potential future research topics are outlined next.

6.2.1 Pre-Selection Criterion in HSAMA

In the HSAMA optimization framework, local searches are applied only to $\eta\%$ individuals of the EA population that were predicted to be elite. Other alternative pre-selection criteria for identifying the promising individuals may be considered.

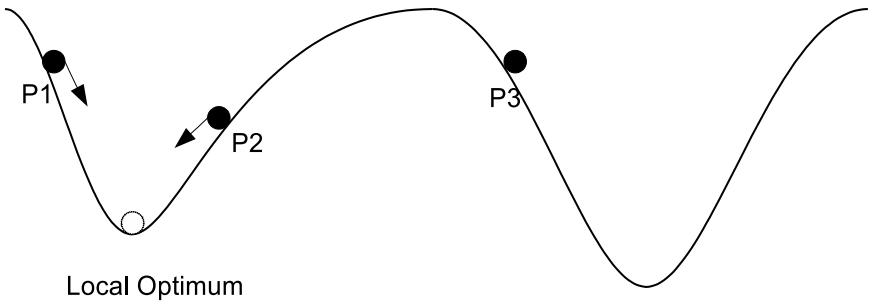


Figure 6.1: Redundant individuals within the same basin of a local minimum

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

For example, Figure 6.1 illustrates the notion of redundant individuals within the same basin of a local optimum. If P_1, P_2, P_3 are among the EA population, since P_1 and P_2 lies within the same basin of attraction, any one individual among the two would lead to the same local optimum through a local search. Hence the further cost saving can be achieved in HSAMA. Then the ultimate challenge is how to identify the redundant individuals within the same basin of attraction.

6.2.2 Enhancement on MSAMA

Here, two potential future directions are described for enhancing the proposed MSAMA optimization framework.

Firstly, the idea of using surrogate ensemble was reported in [69], where the neural network ensemble instead of a single neural network is used to improve the approximation accuracy during fitness evaluation. Hence, the use of surrogate ensemble represents a potential way to mitigate the effect of '*curse of uncertainty*' in the MSAMA, while multiple surrogate models or ensembles may also provide the benefit of '*blessing of uncertainty*'.

Secondly, novel adaptive strategies may be introduced for predicting the most suitable local surrogate model(s) among the several diverse surrogate models built in MSAMA for conducting local searches. The ultimate challenge here is how to predict the best local surrogate model(s) among the multiple surrogate models employed.

6.2.3 Analysis of Global Convergence Rates

Global convergence is defined in the optimization literature as the mathematical assurance that the iterates produced by an algorithm, started from an arbitrary initial guess, will converge to a global optimum of the original objective function.

Rudolph [101] analyzes the convergence properties of the standard GA with mutation, crossover and proportional reproduction applied to static optimization problems. The

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

global convergence analyses of surrogate-assisted memetic algorithms are also analyzed in [8], [25].

However, there exists a tradeoff between the global convergence rate and the long-term performance of the algorithms as reported in [102]. That is to say, the larger the mutation probability is and the smaller the population size and the encoding length are, the more quickly the standard GA converges. On the other hand, it is well recognized that these GA parameters have almost the contrary effects on the long-term performance of the standard GA.

To achieve the tradeoff between the global convergence rate and the long-term performance of the proposed SAMAs, theoretical analysis of global convergence rates for the proposed SAMAs is an excellent future research direction.

6.2.4 Grid-enabled SAMA variants

Due to large design spaces in CED optimization problems, computational budget based on a single cluster may be limited. Further, the use of Grid computing technologies provides abundant computational resources that could benefit evolutionary optimization. Hence, by migrating the most time consuming portion of the design cycle onto a Grid computing environment, i.e., through creating a Grid-enabled parallel optimization frameworks [103], would help shorten the entire design cycle time. For example, specialized analysis codes possessed by different design teams that spans across geographically distributed locations may be shared and better utilized.

Grid-enabled proposed SAMA variants for solving computationally expensive optimization problems can significantly shorten the design cycle by utilizing the powerful grid resources. To attain immense parallelism in the evolutionary search and seamless access to Grid resources, the effective scheduling algorithm and error recovery mechanism for exact fitness evaluation of individuals on heterogenous Grid resources represents some challenging future research direction.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

6.2.5 Effect of the Genetic Operators

The present dissertation work has focused solely on the use of meta-modeling techniques in the context of evolutionary computation. Nevertheless, it is worth noting that the search performance of any evolutionary algorithm including HSAMA and MSAMA will be greatly affected by the configurations of the chosen variation operators. Although the evolutionary operators were configured based on rule of thumb that have been shown to work well in this dissertation work, a further research work would clearly be to study and analyze the impact of the operators on SAEA search.

6.2.6 High Dimensionality Optimization

The computational complexity of the proposed SAMAs depends on the choice of approximation methods used for creating the surrogate models. Further, it is well-known that the computational complexity also increases with dimensionality because increasing number of training data points would be needed to build reasonable accurate surrogate model due to the effect of *'curse of dimensionality'*. Hence, it may not be possible to easily generalize the computational complexity of the proposed SAMAs.

More importantly, it is worth noting that, such computational complexity for creating the surrogate model proves to be insignificant in the context of computationally expensive optimization problems with fitness function evaluations that may cost several hours or days to compute a single evaluation. Nevertheless, study on SAMAs for handling very high dimensionality problems remains to be a big challenge.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

List of Publications

Journal

(i) Zongzhao Zhou, Yew Soon Ong, Prasanth B. Nair, Andy J Keane and Kai Yew Lum, "Combining Global and Local Surrogate Models to Accelerate Evolutionary Optimization," *IEEE Transaction on Systems, Man and Cybernetics (SMC), part C - Applications and Reviews*, vol. 37, no. 1, pp. 66-76, 2007.

(ii) Zongzhao Zhou, Yew Soon Ong, Meng Hiot Lim and Bu Sung Lee, "Memetic Algorithm using Multi-Surrogates for Computationally Expensive Optimization Problems," *Soft Computing*, vol. 11, no. 10, pp. 957-971, 2006.

Conference

(i) Yew Soon Ong, Zongzhao Zhou and Dudy Lim, "Curse and Blessing of Uncertainty in Evolutionary Algorithm Using Approximation," *In Proc. IEEE World Congress on Computational Intelligence (WCCI2006)*, pp. 2928-2935, British Columbia, Canada, July 2006.

(ii) Zongzhao Zhou, Yew Soon Ong, My Hanh Nguyen and Dudy Lim, "A Study on Polynomial Regression and Gaussian Process Global Surrogate Model in Hierarchical Surrogate-Assisted Evolutionary Algorithm," *In Proc. IEEE Congress on Evolutionary Computation (CEC'05)*, pp. 2832-2839, Edinburgh, United Kingdom, September 2005.

(iii) Zongzhao Zhou, Yew Soon Ong and Prasanth B. Nair, "Hierarchical Surrogate-Assisted Evolutionary Optimization Framework," *In Proc. IEEE Congress on Evolutionary Computation (CEC'04)*, pp. 1586-1593, Portland, USA, June 2004.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Appendix A: Benchmark Test Problems

Some commonly used benchmark test problems already extensively discussed in the literature are used here in this research work. They represent classes of general constrained, unimodal and multimodal continuous parametric test problems. The unimodal benchmark test problems are bound-constrained Step, Sphere and Rosenbrock functions. The multimodal benchmark test problems are the bound constrained Ackley, Griewank and Rastrigin functions. All the benchmark test problems considered have diverse fitness landscapes for study.

A.1 Step Test Function

The Step function consists of flat plateaus with $slope = 0$ in an underlying continuous function. It is hard to find the global optimum because minor changes of the object variables do not affect the fitness. Therefore no conclusions about the search direction are possible. It is defined as [104]:

$$f(\mathbf{x}) = \sum_{i=1}^{n} \lfloor x_i^2 \rfloor \tag{A.1}$$
$$-5.12 \leq x_i \leq 5.12, i = 1, 2, \ldots, n.$$

where $\mathbf{x} \in \mathbb{R}^n$ is the vector of design variables, and n is the variables size, respectively. $\lfloor . \rfloor$ represents the floor function. The 2-D surface of this function for [-2, 2] is shown in Figure A.1.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

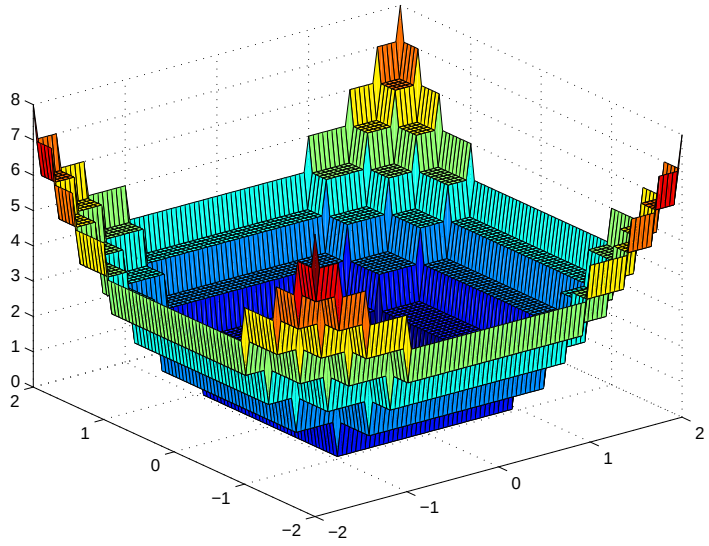


Figure A.1: 2-D Step benchmark test function

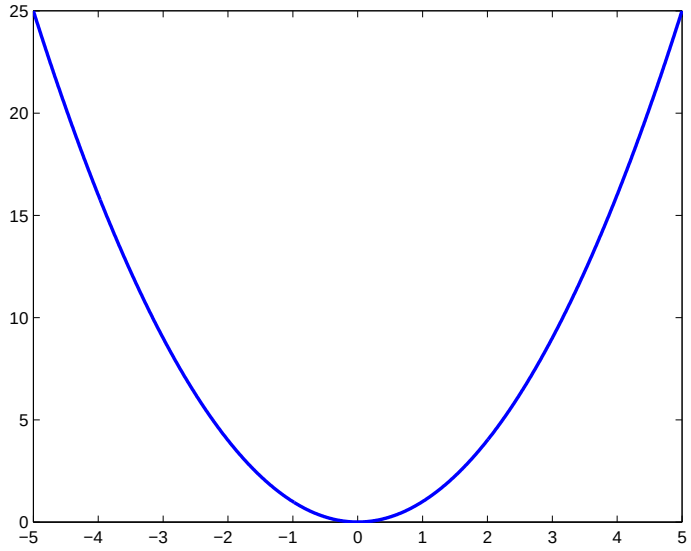


Figure A.2: 1-D Sphere benchmark test function

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

A.2 Sphere Test Function

The Sphere test problem is a smooth, symmetric function and is used to provide a measure of the general eciency of the proposed strategy. It has a single minimum of zero located at $(0, \ldots, 0)$. The function is defined as [104]:

$$f(\mathbf{x}) = \sum_{i=1}^{n}(x_i^2) \tag{A.2}$$
$$-5.12 \leq x_i \leq 5.12, i = 1, 2, \ldots, n.$$

where $\mathbf{x} \in \mathbb{R}^n$ is the vector of design variables, and n is the variables size, respectively. Figure A.2 shows a 1-dimensional Sphere function for [-5, 5].

A.3 Rosenbrock Test Function

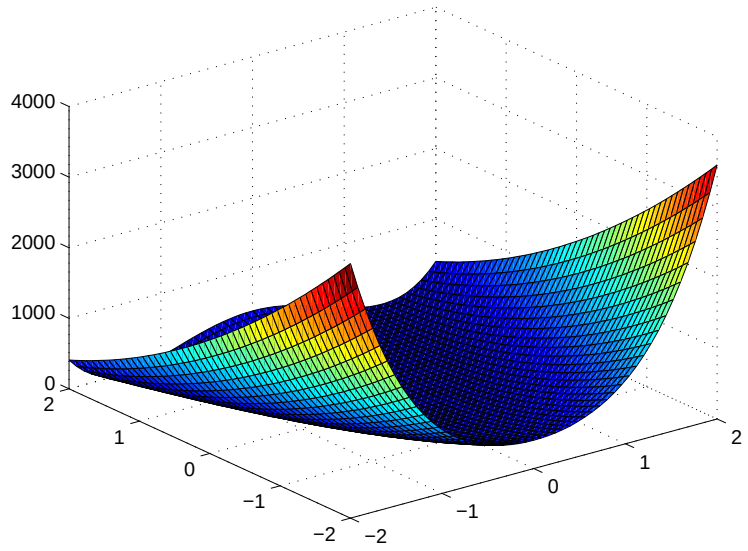


Figure A.3: 2-D Rosenbrock benchmark test function

Rosenbrock test function is nonseparable, with highly correlated decision variables. As an added difficulty the minimum of zero is located at $(1, \ldots, 1)$ in a long, flat-bottomed, curved, and narrow valley. It is defined as [104]:

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

$$f(\mathbf{x}) = \sum_{i=1}^{n-1}(100 \times (x_{i+1} - x_i^2)^2 + (1 - x_i)^2) \tag{A.3}$$
$$-2.048 \leq x_i \leq 2.048, i = 1, 2, \ldots, n-1.$$

where $\mathbf{x} \in \mathbb{R}^n$ is the vector of design variables, and n is the variables size, respectively. The surface of this function for $n = 2$ is shown in Figure A.3.

This function is claimed to be unimodal in a number of past studies. However, for $n > 3$, this function has more than one minima, which is identified in [105].

A.4 Ackley Test Function

Ackley test function is symmetric and very bumpy. Its number of local minima increases exponentially with the problem dimension. It has a global minimum of zero located at $(1, \ldots, 1)$ with very strong local features. It is defined as [18]:

$$f(\mathbf{x}) = 20 + e - 20e^{-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}} - e^{\frac{1}{n}\sum_{i=1}^{n}\cos(2\pi x_i)} \tag{A.4}$$
$$-32.768 \leq x_i \leq 32.768, i = 1, 2, \ldots, n.$$

where $\mathbf{x} \in \mathbb{R}^n$ is the vector of design variables, and n is the variables size, respectively. The surface of the Ackley function for $n = 2$ is shown in Figure A.4.

A.5 Griewank Test Function

The Griewank test problem is a high dimensional multimodal function with many local minima and a global minimum of zero located at $(0, \ldots, 0)$. It is defined as [18]:

$$f(\mathbf{x}) = 1 + \sum_{i=1}^{n} x_i^2/4000 - \prod_{i=1}^{n}\cos(x_i/\sqrt{i}) \tag{A.5}$$
$$-600 \leq x_i \leq 600, i = 1, 2, \ldots, n.$$

where $\mathbf{x} \in \mathbb{R}^n$ is the vector of design variables, and n is the variables size, respectively. Figure A.5 shows a one-dimensional slice of this function for [-200, 200].

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

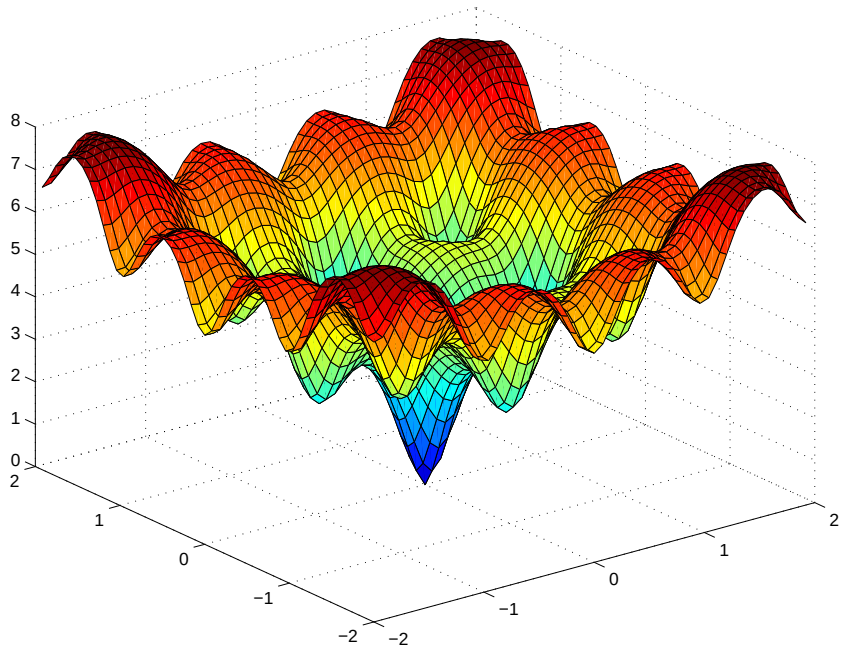


Figure A.4: 2-D Ackley benchmark test function

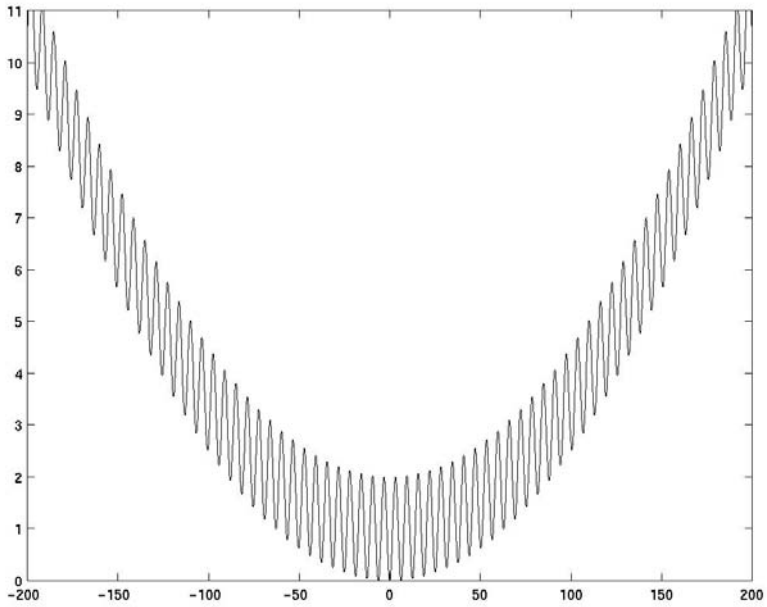


Figure A.5: 1-D Griewank benchmark test function

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

This function has inter-parameter linkage due to the presence of the product term. However, the effect decreases as the number of parameters increases. The Griewank function with 10 dimensions, $n = 10$, has more than 500 local minima in the hybercube $[-600, 600]^{10}$. It has a very rugged landscape and is difficult to optimize [18].

A.6 Rastrigin Test Function

The Rastrigin test problem is also a high dimensional multimodal function with many local minima and a global minimum of zero located at $(0, \ldots, 0)$. It also has a very rugged landscape and is difficult to optimize.

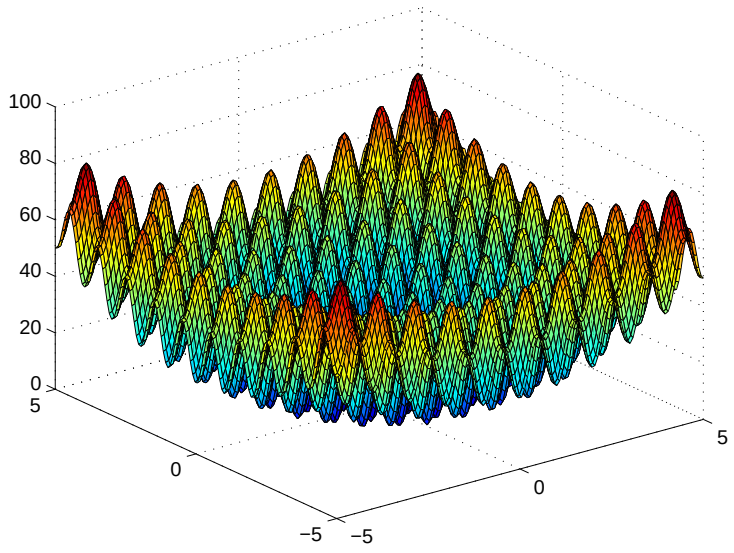


Figure A.6: 2-D Rastrigin benchmark test function

The bound constrained Rastrigin function is defined as [18]:

$$f(\mathbf{x}) = 10n + \sum_{i=1}^{n}(x_i^2 - 10\cos(2\pi x_i)) \qquad (A.6)$$
$$-5.12 \leq x_i \leq 5.12, i = 1, 2, \ldots, n.$$

where $\mathbf{x} \in \mathbb{R}^n$ is the vector of design variables, and n is the variables size, respectively. The surface of the Rastrigin function for $n = 2$ is shown in Figure A.6.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Appendix B: Aerodynamic Shape Design Optimization Problem

The adjoint method has been applied since the early 1990's to the problem of aerodynamic shape design optimization and is based on optimal control theory applied to systems governed by partial differential equations. The variables of Aerodynamic shape design are the flow variables and shape parameters, and where the objective is some aerodynamic performance functional. For differentiable shape representations, this method is very efficient as it is essentially a steepest-descent method, where the gradient is obtained analytically by solving the adjoint equation. For multi-modal problems, however, this method tends to arrive at only a local minimum.

The aerodynamic shape design optimization problem is also used as a real application in this research work. In particular, the parametric design optimization of a 2-D airfoil structure with minimum drag-over-lift ratio, i.e. D/L is considered. The drag D and lift L on an airplane are the components of the total aerodynamic force parallel and vertical to the direction of flight, respectively, as shown in Fig. B.1.a. The importance of the D/L ratio in design can be understood, for example, in two airplane performance considerations [106]. First, the engine thrust required for level and unaccelerated flight, i.e. T_{cruise}, is given by

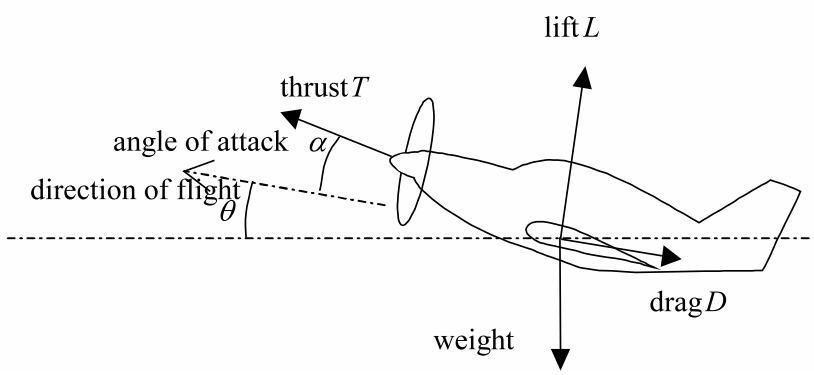
$$T_{cruise} = (weight\ of\ aircraft) \times D/L \tag{B.1}$$

Second, an airplane in a power-off gliding flight will descend at an angle - $\theta_{gliding}$ gliding given by

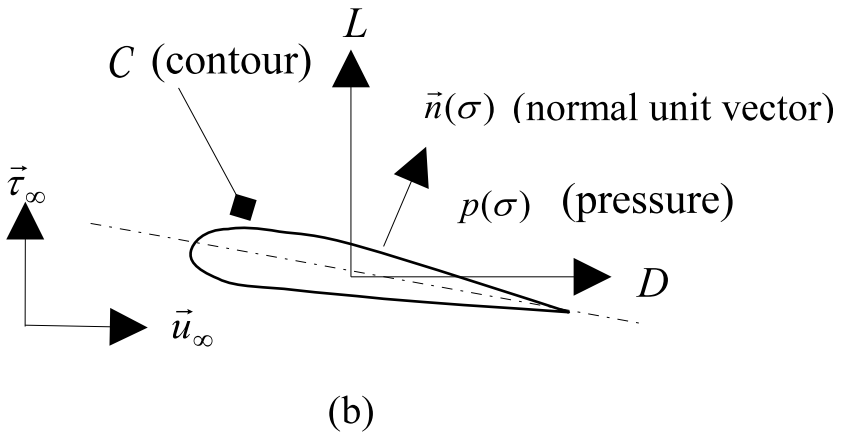
$$\tan\theta_{gliding} = D/L \tag{B.2}$$

In both cases, it is obvious that the smaller the ratio D/L, the better the performance. In the first case, a small ratio means less engine power is required for cruising flight, thus

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library



B.1.a: An airplane.



B.1.b: An airfoil.

Figure B.1: Forces acting on an airplane and airfoil.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

saving fuel. In the second case, low drag over lift entails a safer gliding flight in the case of engine failure. While the drag and lift forces on an airplane are determined by various body components, the contribution of the wings is dominant. This motivates the development of an approach for designing airfoil geometries by minimizing the D/L ratio.

In an airfoil shape optimization problem using computation fluid dynamics, the drag and lift forces can be obtained by calculating the flow field around the airfoil under prescribed operating conditions, defined by the Mach number which represents the incident flow rate, and the angle of attack (see Fig. B.1.a). Ignoring friction, the flow is governed by the 2D Euler equations:

$$\frac{\partial w}{\partial t} + \frac{\partial f_1}{\partial z_1} + \frac{\partial f_2}{\partial z_2} = 0 \tag{B.3}$$

with t as the time variable,

$$w = \begin{bmatrix} \rho \\ \rho u_1 \\ \rho u_2 \\ \rho E \end{bmatrix}, f_1 = \begin{bmatrix} \rho u_1 \\ \rho u_1^2 + p \\ \rho u_1 u_2 \\ \rho u_1 H \end{bmatrix}, f_2 = \begin{bmatrix} \rho u_2 \\ \rho u_1 u_2 \\ \rho u_2^2 \\ \rho u_2 H \end{bmatrix} \tag{B.4}$$

where ρ is the fluid density, u_1 and u_2 are the flow velocity components in the Cartesian space with coordinates z_1 and z_2, p is the fluid static pressure, E is the total specific energy and H is the total specific enthalpy. Moreover, the pressure is given by $p = (\gamma - 1)\rho(E - \frac{1}{2}u_1^2 - \frac{1}{2}u_2^2)$, where γ is the ratio of the specific heat [107].

Thus, the drag D and lift L are simply the components opposite the direction of flight $\vec{u}_\infty$, and the direction perpendicular to flight $\vec{\tau}_\infty$, respectively, of the resultant force due to pressure acting along the contour $\mathcal{C}$ of the airfoil (see Fig. B.1.b). Let σ denote differentiation with respect to the position along the streamwise direction. They are given by the following integrals:

$$D = \oint_{\mathcal{C}} p(\sigma)\vec{n}(\sigma).\vec{u}_\infty d\sigma \tag{B.5}$$

and

$$L = \oint_{\mathcal{C}} p(\sigma)\vec{n}(\sigma).\vec{\tau}_\infty d\sigma. \tag{B.6}$$

The optimization problem considered here is to achieve an airfoil design for an optimized drag-to-lift ratio profile for constant operating conditions of Mach 0.5 and Angle

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

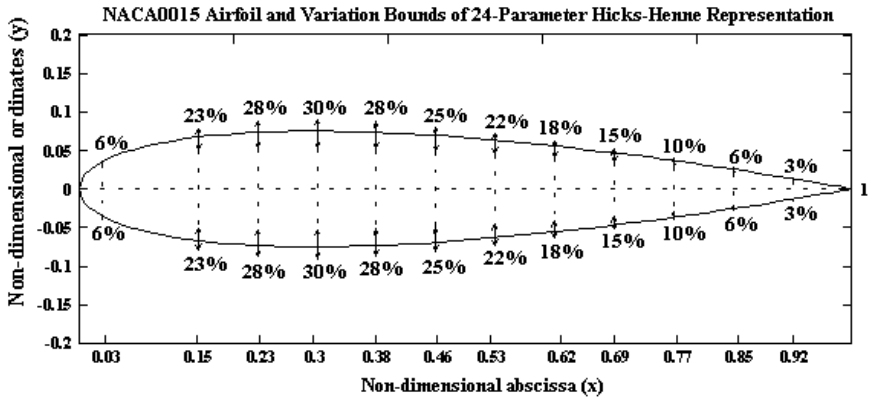


Figure B.2: Airfoil geometry characterized using 24-parameter Hicks-Henne functions.

of Attack, $AOA = 2.0$ degrees. The geometry of the airfoil is represented using 24-parameter Hicks-Henne functions [108] as illustrated in Figure B.2.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Appendix C: Lamarckian and Baldwinian Learning

The memetic algorithm is a class of stochastic heuristics for global optimization which combine the parallel global search nature of evolutionary algorithm with local search to improve individual solutions. This technique is being applied to an increasing range of application domains with successful results. Two basic learning strategies are typically considered in MAs, i.e, Lamarckian and Baldwinian learning

A common form of hybrid genetic algorithm uses local searh to improve the initial population as well as the genetic strings produced by genetic recombination. The resulting improvements are then coded onto the strings processed by the genetic algorithm. This is equivalent to a form of lamarckian evolution/learning.

Local search in this context can be thought of as being analogous to a kind of learning that occurs during the lifetime of an individual string. But there is another way in which learning (i.e., local search) and evolution can interact. Instead of coding the improvements back onto the string, the fitness value of the improvement can be transferred to the individual. This has the effect of changing the fitness landscape, but the resulting form of evolution is still "Darwinian" in nature. Various phenomena associated with this form of combining learning and evolution are collectively known as Baldwinian learning.

In summary, MA is referred as Lamarckian if the outcome of the local search process (both the genetic string and fitness value) replaces the original population, and Baldwinian if the original genetic string is kept, while its fitness value is replaced with the locally improved fitness value of local search process.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

References

[1] C. T. Lawrence and A. L. Tits, "Nonlinear equality constraints in feasible sequential quadratic programming," *Optimization Methods and Software*, vol. 6, pp. 265–282, 1996.

[2] H.-P. Schwefel, *Evolution and Optimum Seeking*. John Wiley & Sons Press, 1995.

[3] J. N. Siddall, *Optimal Engineering Design: Principles and Applications*. Marcel Dekker, Inc., 1982.

[4] V. C. Chen, K.-L. Tsui, R. R. Barton, and M. Meckesheimer, "A review on design, modeling and applications of computer experiments," *IIE Transactions*, vol. 38, no. 4, pp. 273–291, 2006.

[5] A. Choudhury, P. B. Nair, and A. J. Keane, "A data parallel approach for large-scale gaussian process modeling," in *Proc. the Second SIAM International Conference on Data Mining*, (Arlington, VA), Apr. 2002.

[6] K. C. Giannakoglou, "Design of optimal aerodynamic shapes using stochastic optimization methods and computational intelligence," *International Review Journal Progress in Aerospace Sciences*, vol. 38, no. 5, pp. 43–76, 2002.

[7] M. K. Karakasis, A. P. Giotis, and K. C. Giannakoglou, "Inexact information aided, low-cost, distributed genetic algorithms for aerodynamic shape optimization," *International Journal for Numerical Methods in Fluids*, vol. 43, pp. 1149–1166, 2003.

[8] Y. S. Ong, P. B. Nair, and A. J. Keane, "Evolutionary optimization of computationally expensive problems via surrogate modeling," *American Institute of Aeronautics and Astronautics Journal*, vol. 41, no. 4, pp. 687–696, 2003.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

[9] A. Ratle, "Accelerating evolutionary algorithms using fitness function models," *Parallel Problem Solving from Nature*, vol. 5, pp. 87–96, 1998.

[10] W. B. Song, *Shape Optimisation Of Turbine Blade Firtrees*. PhD thesis, Univ. of Southampton, 2002.

[11] G. G. Wang and S. Shan, "Review of metamodeling techniques in support of engineering design optimization," *Journal of Mechanical Design*, vol. 129, no. 4, pp. 370–380, 2007.

[12] I. Foster, J. N. C. Kesselman, and S. Tuecke, "The physiology of the grid: An open grid services architecture for distributed systems integration," tech. rep., Open Grid Service Infrastructure WG, Global Grid Forum, June 2002.

[13] I. Foster and C. Kesselman, eds., *The Grid: Blueprint for a New Computing Infrastructure*. San Francisco, USA: Morgan Kaufmann Publishers, 1999.

[14] I. Foster, C. Kesselman, and S. Tuecke, "The anatomy of the grid: Enabling scalable virtual organizations," *International J. Supercomputer Applications*, vol. 15, no. 3, 2001.

[15] A. J. Keane, "Experiences with optimizers in structural design," in *Proc. the First International Conference on Adaptive Computing in Engineering Design and Control*, (Plymouth, UK), pp. 14–27, Sept. 1994.

[16] D. Whitley, "A genetic algorithm tutorial," *Statistics and Computing*, vol. 4, pp. 65–85, 1994.

[17] J. H. Holland, *Adaptation in Natural and Artificial Systems*. MIT Press, 1975.

[18] A. Torn and A. Zilinskas, *Global Optimization*. Lecture Notes in Computer Science, Vol. 350, Springer-Verlag, 1989.

[19] W. E. Hart, *Adaptive Global Optimization with Local Search*. PhD thesis, Univ. of California, San Diego, CA, 1994.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

[20] P. Moscato, "On evolution, search, optimization, genetic algorithms and martial arts: Towards memetic algorithms," Tech. Rep. C3P 826, California Institue of Technology, Pasadena, CA, 1989.

[21] L. Davis, *Handbook of Genetic Algorithms*. Van Nostrand Reinhold, New York, 1991.

[22] Y. S. Ong and A. Keane, "Meta-lamarckian in memetic algorithm," *IEEE Transactions On Evolutionary Computation*, vol. 8, no. 2, pp. 99–110, 2004.

[23] R. Dawkins, *The Selfish Gene*. Oxford University, 1976.

[24] D. L. Whitley, V. S. Gordon, and K. E. Mathias, "Lamarckian evolution, the baldwin effect and function optimization," in *Parallel Problem Solving from Nature – PPSN III* (Y. Davidor, H.-P. Schwefel, and R. Männer, eds.), (Berlin), pp. 6–15, Springer, 1994.

[25] Y. S. Ong, M. H. Lim, N. Zhu, and K. W. Wong, "Classification of adaptive memetic algorithms: A comparative study," *IEEE Transactions On Systems, Man and Cybernetics - Part B*, vol. 36, no. 1, pp. 141–152, 2006.

[26] B. D. Bunday, *Basic Optimization Methods*. Edward Arnold (Publishers) Ltd, 1985.

[27] C. Kelley, *Solving Nonlinear Equations with Newton's Method*. SIAM Press, 2003.

[28] A. J. Booker, J. E. J. Dennis, P. D. Frank, D. B. Serafini, V. Torczon, and M. W. Trosset, "A rigorous framework for optimization of expensive functions by surrogates," *Structural Optimization*, vol. 17, no. 1, pp. 1–13, 1998.

[29] R. A. Canfield, "Multi-point cubic surrogate functions for sequential approximate optimization," *Journal of Structural and Multidisciplinary Optimization*, vol. 27, no. 5, pp. 323–336.

[30] D. B. Serafini, *A Framework for Managing Models in Nonlinear Optimization of Computationally Expensive Functions*. PhD thesis, Rice University, 1998.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

[31] T. W. Simpson, "A comparison of metamodeling strategies for computer-based engineering design," in *INFORMS Philadelphia Fall 1999 Meeting*, (Philadelphia, USA), Nov. 1999.

[32] T. W. Simpson, J. Peplinski, P. N. Koch, and J. K. Allen, "Metamodels for computer-based engineering design: Survey and recommendations," *Engineering with Computers*, vol. 17, no. 2, pp. 129–150, 2001.

[33] T. W. Simpson, A. J. Booker, D. Ghosh, A. A. Giunta, P. N. Koch, and R. J. Yang, "Approximation methods in multidisciplinary analysis and optimization: A panel discussion," *Structural and Multidisciplinary Optimization*, vol. 27, no. 5, pp. 302–313, 2004.

[34] H. Furuya and R. Haftka, "Combining genetic and deterministic algorithms for locating actuators on space structures," *Journal of Spacecraft and Rockets, AIAA-95-1400*, vol. 33, no. 3, pp. 422–427, 1996.

[35] P. B. Nair and A. J. Keane, "Improved genetic algorithms for the design of stiffened composite panels," *Computers and Structures*, vol. 58, no. 3, pp. 543–555, 1996.

[36] G. M. Robinson and A. J. Keane, "A case for multi-level optimisation in aeronautical design," in *Proc. the RAeS Conference on Multidisciplinary Design and Optimisation*, (The Royal Aeronautical Society), pp. 9.1 – 9.6, Sept. 1998.

[37] K. H. Liang, X. Yao, and C. Newton, "Evolutionary search of approximated n-dimensional landscapes," *International Journal of Knowledge-Based Intelligent Engineering Systems*, vol. 4, no. 3, pp. 172–183, 2000.

[38] D. R. Jones, M. Schonlau, and W. J. Welch, "Efficient global optimization of expensive black-box functions," *Journal of Global Optimization*, vol. 13, pp. 455–492, 1998.

[39] G. E. P. Box and N. R. Draper, *Empirical model-building and response surface.* Wiley Press, 1987.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

[40] T. W. Simpson, J. D. Peplinski, P. N. Koch, and J. K. Allen, "On the use of statistics in design and the implications for deterministic computer experiments," in *Design Theory and Methodology - DTM'97*, (Sacramento, CA), pp. 35–45, Sept. 1997. ASME, Paper No. DETC97/DTM-3881.

[41] V. Vapnik, *Statistical Learning Theory*. John Wiley and Sons, 1998.

[42] C. Bishop, *Neural Networks for Pattern Recognition*. Oxford University Press, 1995.

[43] R. H. Myers and D. C. Montgomery, *Response Surface Methodology: Process and Product Optimization Using Designed Experiments*. Wiley & Sons, 1995.

[44] T. W. Simpson, T. M. Mauery, J. J. Korte, and F. Mistree, "Kriging metamodels for global approximation in simulation-based multidisciplinary design optimization," *AIAA Journal*, vol. 39, no. 12, pp. 2233–2241, 2001.

[45] J. Sacks, W. J. Welch, T. J. Mitchell, and H. P. Wynn, "Design and analysis of computer experiments," *Concurrency Computation Practice and Experience*, vol. 4, no. 4, pp. 409–435, 1989.

[46] C. K. I. Williams and C. E. Rasmussen, *Gaussian Processes for Regression*. MIT Press, 1996.

[47] J. H. Friedman, "Multivariate adaptive regression splines," *Annals of Statistics*, vol. 19, no. 1, pp. 1–67, 1991.

[48] T. W. Simpson, T. M. Mauery, J. J. Korte, and F. Mistree, "Comparison of response surface and kriging models for multidisciplinary design optimization," in *Proc. the 7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, (St. Louis, USA), pp. 10–31, 1998. AIAA Paper 98-4758.

[49] A. A. Guinta and L. Watson, "A comparison of approximation modelling techniques: polynomial versus interpolating models," in *Proc. the 7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, vol. 1, (St. Louis, USA), pp. 392–404, AIAA–98–4758, Sept. 1998.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

[50] D. D. Daberkow and D. N. Marvis, "New approaches to conceptual and preliminary aircraft design: A comparative assessment of a neural network formulation and a response surface methodology," in *Proc. 1998 World Aviation Conference, AIAA*, (Anaheim, USA), pp. 866–878, Sept. 1998, AIAA-98-5509.

[51] S. Varadarajan, W. Chen, and C. Pelka, "The robust concept exploration method with enhanced model approximation capabilities," *Engineering Optimization*, vol. 32, no. 3, pp. 309–334, 2000.

[52] R. Jin, W. Chen, and T. W. Simpson, "Comparative studies of metamodeling techniques under multiple modeling criteria," *Structural and Multidisciplinary Optimization*, vol. 23, no. 1, pp. 1–13, 2001.

[53] T. W. Simpson, D. K. J. Lin, and W. Chen, "Sampling strategies for computer experiments: Design and analysis," *International Journal of Reliability and Applications*, vol. 2, no. 3, pp. 209–240, 2001.

[54] R. Vaidyanathan, K. P. Tucker, N. Papila, and W. Shyy, "CFD based design optimization for a single element rocket injector," *Journal of Propulsion and Power*, vol. 30, no. 4, pp. 705–717, 2004.

[55] N. V. Queipo, R. T. Haftka, W. Shyy, T. Goel, R. Vaidyanathan, and P. Tucker, "Surrogate-based analysis and optimization," *Progress in Aerospace Sciences*, vol. 10, no. 41, pp. 1–28, 2005.

[56] Y. Jin and J. Branke, "Evolutionary optimization in uncertain environments: A survey," *IEEE Transactions on Evolutionary Computation*, vol. 9, no. 3, pp. 303–317, 2005.

[57] K. Rasheed, "Informed operators: Speeding up genetic-algorithm-based design optimization using reduced models," in *Proceedings of Genetic and Evolutionary Computation Conference*, (Las Vegas, USA), pp. 628–635, Morgan Kaufmann, 2000.

[58] K. Anderson and Y. Hsu, "Genetic crossover strategy using an approximation concept," in *IEEE Congress on Evolutionary Computation*, (Washington D.C.), pp. 527–533, 1999.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

[59] A. Mutoh, S. Kato, T. Nakamura, and H. Itoh, "Reducing execution time on genetic algorithms in real-world applications using fitness prediction," in *Proc. of the IEEE Congress on Evolutionary Computation*, vol. 1, (Sidney, Australia), pp. 552–559, IEEE, 2003.

[60] K. Rasheed, S. Vattam, and X. Ni, "Comparison of methods for using reduced models to speed up design optimization," in *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO)*, (New York, USA), pp. 1180–1187, 2002.

[61] K. A. M. Schoenauer, "Surrogate deterministic mutation," in *Artificial Evolution'01*, pp. 103–115, Springer, 2002.

[62] D. Büche, N. N. Schraudolph, and P. Koumoutsakos, "Accelerating evolutionary algorithms with gaussian process fitness function models," *IEEE Transactions on Systems, Man, and Cybernetics, Part C, Special Issue on Knowledge Extraction and Incorporation in Evolutionary Computation*, vol. 35, pp. 183–194, 2005.

[63] M. A. El-Beltagy, P. B. Nair, and A. J. Keane, "Metamodelling techniques for evolutionary optimization of computationally expensive problems: Promise and limitations," in *Proc. IEEE Genetic and Evolutionary Computation Conference (GECCO'99)*, (Florida , USA), pp. 196–203, July 1999.

[64] A. Ratle, "Kriging as a surrogate fitness landscape in evolutionary optimization," *Artificial Intelligence for Engineering Design Analysis and Manufacturing*, vol. 15, no. 1, pp. 37–49, 2001.

[65] H. Ulmer, F. Streichert, and A. Zell, "Evolution startegies assisted by gaussian processes with improved pre-selection criterion," in *Proc. IEEE Congress on Evolutionary Computation CEC'03*, (Canberra, Australia), pp. 1741–1751, Dec. 2003.

[66] Y. S. Ong, P. B. Nair, and K. Y. Lum, "Max-min surrogate-assisted evolutionary algorithm for robust aerodynamic design," *IEEE Transaction on Evolutionary Computation*, vol. 10, no. 4, pp. 392–404, 2006.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

[67] M. Emmerich, A. Giotis, M. Ozdemir, T. Back, and K. Giannakoglou, "Metamodel-assisted evolution strategies," in *Proc. the 7th International Conference on Parallel Problem Solving from Nature (PPSN 2002)*, (Granada, Spain), pp. 361–370, Sept. 2002.

[68] R. G. Regis and C. A. Shoemaker, "Local function approximation in evolutionary algorithms for the optimization of costly functions," *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 5, pp. 490 – 505, 2004.

[69] Y. Jin and B. Sendhoff, "Reducing fitness evaluations using clustering techniques and neural networks ensembles," in *Proc. the Genetic and Evolutionary Computation Conference*, (Seattle, USA), pp. 688–699, June 2004.

[70] P. Nain and K. Deb, "Computationally effective search and optimization procedure using coarse to fine approximation," in *Proc. Congress on Evolutionary Computation (CEC'03)*, (Canberra, Australia), pp. 2081–2088, Dec. 2003.

[71] J. Xuan, D. Chafekar, and K. Rasheed, "Constrained multi-objective ga optimization using reduced models," in *Proc. the Genetic and Evolutionary Computation Conference (GECCO'03)*, (Chicago, USA), pp. 174–177, July 2003.

[72] J. Knowles, "ParEGO: A hybrid algorithm with on-line landscape approximation for expensive multiobjective optimization problems," *IEEE Transactions on Evolutionary Computation*, vol. 10, no. 1, pp. 50–66, 2006.

[73] Y. Jin, "A comprehensive survey of fitness approximation in evolutionary computation," *Soft Computing*, vol. 9, no. 1, pp. 3–12, 2005.

[74] Y. Jin, "Fitness approximation in evolutionary computation bibliography, http://www.soft-computing.de/amec.html," 2007.

[75] Y. S. Ong, P. B. Nair, A. J. Keane, and K. W. Wong, "Surrogate-assisted evolutionary optimization frameworks for high-fidelity engineering design problems," in *Knowledge Incorporation in Evolutionary Computation, Studies in Fuzziness and Soft Computing* (Y. Jin, ed.), pp. 307–332, Springer Verlag, 2004.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

[76] Y. Jin, M. Olhofer, and B. Sendhoff, "Managing approximate models in evolutionary aerodynamic design optimization," in *Proc. IEEE the Genetic and Evolutionary Compution Conference (GECCO'01)*, (San Francisco, USA), pp. 592–599, May 2001.

[77] Y. Jin, M. Olhofer, and B. Sendhoff, "On evolutionary optimization with approximate fitness functions," in *Proc. IEEE the Genetic and Evolutionary Compution Conference (GECCO'00)*, (Las Vegas, USA), pp. 786–792, July 2000.

[78] K. S. Won, T. Ray, and K. Tai, "Modelling and managing approximate functions within an optimization framework," in *Proc. the 2nd International Conference on Computational Intelligence, Robotics and Autonomous Systems*, (Singapore), Dec. 2003. paper no. PS02-05-02.

[79] M. Salami and T. Hendtlass, "A fast evaluation strategy for evolutionary algorithms," *Applied Soft Computing*, vol. 2, pp. 156–173, 2003.

[80] J. Branke and C. Schmidt, "Fast convergence by means of fitness estimation," *Soft Computing Journal*, vol. 9, no. 1, pp. 13–20, 2005.

[81] M. A. El-Beltagy and A. J. Keane, "Evolutionary optimization for computationally expensive problems using gaussian processes," in *Proc. the International Conference on Artificial Intelligence (IC-AI'2001)s* (H. Arabnia, ed.), vol. 2, (Las Vegas, USA), pp. 708–714, CSREA Pres, 2001.

[82] V. Torczon and M. W. Trosset, "Using approximations to accelerate engineering design optimization," in *Proc. the 7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, AIAA Paper 98-4800, TR-98-33*, (St. Louis, USA), Sept. 1998.

[83] M. Emmerich, K. C. Giannakoglou, and B. Naujoks, "Single-and multi-objective optimization assisted by gaussian random field metamodels," *IEEE-TEC Special Issue on Evolutionary Computation in the presence of uncertainty*, vol. 10, no. 4, pp. 421–439, 2006.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

[84] M. Bhattacharya and G. Lu, "Dafhea: A dynamic approximate fitness based hybrid ea for optimization problems," in *Proceedings of IEEE Congress on Evolutionary Computation*, pp. 1879–1886, 2003.

[85] Y. Jin, T. Okabe, and B. Sendhoff, "Adapting weighted aggregation for multiobjective evolution strategies," in *Proc. the First International Conference on Evolutionary Multi-Criterion Optimization*, (Zurich, Switzerland), pp. 96–110, Springer-Verlag, Mar. 2001. Lecture Notes in Computer Science No. 1993.

[86] Y. Jin, M. Olhofer, and B. Sendhoff, "A framework for evolutionary optimization with approximate fitness functions," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 5, pp. 481–494, 2002.

[87] J. Dennis and V. Torczon, "Managing approximate models in optimization," in *Multidisciplinary design optimization: State-of-the-art* (N. Alexandrov and M. Hussani, eds.), pp. 330–347, SIAM, 1997.

[88] N. Alexandrov, J. E. Dennis, R. M. Lewis, and V. Torczon, "A trust region framework for managing the use of approximation models in optimization," *Structural Optimization*, vol. 15, no. 1, pp. 16–23, 1998.

[89] K.-T. Fang, C.-X. Ma, and P. Winker, "Centered l2-discrepancy of random sampling and latin hypercube design,and construction of uniform designs," *Mathematics of Computation*, vol. 71, no. 237, pp. 275–296, 2000.

[90] W. Shyy, P. K. Tucker, and R. Vaidyanathan, "Response surface and neural network techniques for rocket engine injector optimization," Tech. Rep. 99-2455, AIAA, 1999.

[91] C. A. Micchelli, "Interpolation of scattered data: Distance matrices and conditionally positive definite functions," *Constructive Approximation*, vol. 2, no. 1, pp. 11–22, 1986.

[92] J. F. Rodriguez, J. E. Renaud, and L. T. Watson, "Convergence of trust region augmented lagrangian methods using variable fidelity approximation data," *Structural Optimization*, vol. 15, no. 3-4, pp. 141–156, 1998.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

[93] D. E. Goldberg, *Genetic Algorithms in Search, Optimisation, and Machine Learning*. Addison-Wesley, 1989.

[94] J. Lee and P. Hajela, "A computationally efficient feasible sequential quadratic programming algorithm," *Society for Industrial and Applied Mathematics*, vol. 11, no. 4, pp. 1092–1118, 2001.

[95] D. J. C. Mackay, "Introduction to gaussian processes," *Neural Networks and Machine Learning*, vol. 168, pp. 133–165, 1998.

[96] Y. Jin, M. Huesken, and B. Sendhoff, "Quality measures for approximate models in evolutionary computation," in *Proc. GECCO Workshops: Workshop on Adaptation, Learning and Approximation in Evolutionary Computation*, (Chicago, USA), pp. 170–174, July 2003.

[97] D. Mackay, "Bayesian interpolation," *Neural Computation*, vol. 4, no. 3, pp. 415–447, 1992.

[98] F. H. Lesh, "Multi-dimensional least-square polynomial curve fitting," *Communications of ACM*, vol. 2, no. 9, pp. 29–30, 1959.

[99] A. L. Edwards, *An Introduction to Linear Regression and Correlation*. W. H. Freeman, 1976.

[100] H. Ng, D. Lim, Y. S. Ong, B. S. Lee, L. Freund, S. Parvez, and B. Sendhoff, "A multi-cluster grid enabled evolution framework for aerodynamic airfoil design optimization," in *International Conference on Natural Computing (ICNC)*, vol. 3611 of *Lecture Notes in Computer Science*, pp. 1112–1121, Springer Berlin/Heidelberg, 2005.

[101] G. Rudolph, "Convergence analysis of canonical genetic algorithms," *IEEE Transactions on Neural Networks*, vol. 5, no. 1, pp. 96–101, 1994.

[102] Y. Gao, "An upper bound on the convergence rates of canonical genetic algorithms," *Complexity International*, vol. 5, 1998.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

[103] D. Lim, Y. S. Ong, Y. Jin, B. Sendhoff, and B. S. Lee, "Efficient hierarchical parallel genetic algorithms using grid computing," *Future Generation Computer Systems: The International Journal of Grid Computing: Theory, Methods and Applications*, vol. 23, no. 4, pp. 658–670, 2007.

[104] D. Jong, *An analysis of the behaviour of a class of genetic adaptive systems*. PhD thesis, University of Michigan, 1975.

[105] K. Deb, A. Anand, and D. Joshi, "A computationally efficient evolutionary algorithm for real-parameter optimization," *Evolutionary Computation Journal*, vol. 10, no. 4, pp. 371–395, 1992.

[106] A. D. J. Anderson, *Introduction to Flight,* 4^{th} *edition*. McGraw-Hill, 2000.

[107] H. K. Versteeg and W. Malalasekera, *An Introduction to Computational Fluid Dynamics - The Finite Volume Method*. Longman Group Ltd, 1995.

[108] R. M. Hicks and P. A. Henne, "Wing design by numerical optimization," *Journal of Aircraft*, vol. 15, no. 7, pp. 407–412, 1978.