The copyright © of this thesis belongs to its rightful author and/or other copyright owner. Copies can be accessed and downloaded for non-commercial or learning purposes without any charge and permission. The thesis cannot be reproduced or quoted as a whole without the permission from its rightful owner. No alteration or changes in format is allowed without permission from its rightful owner.

NON-WEIGHTED AGGREGATE EVALUATION FUNCTION OF MULTI-OBJECTIVE OPTIMIZATION FOR KNOCK ENGINE MODELING

AZHER RAZZAQ HADI WITWIT

Universiti Utara Malaysia

DOCTOR OF PHILOSOPHY
UNIVERSITI UTARA MALAYSIA
2017

Awang Had Salleh
Graduate School
of Arts And Sciences

Universiti Utara Malaysia

PERAKUAN KERJA TESIS / DISERTASI
(Certification of thesis / dissertation)

Kami, yang bertandatangan, memperakukan bahawa
(We, the undersigned, certify that)

AZHER RAZZAQ HADI WITWIT

calon untuk Ijazah PhD
(candidate for the degree of)

telah mengemukakan tesis / disertasi yang bertajuk:
(has presented his/her thesis / dissertation of the following title):

"NON-WEIGHTED AGGREGATE EVALUATION FUNCTION OF MULTI-OBJECTIVE OPTIMIZATION FOR KNOCK ENGINE MODELING"

seperti yang tercatat di muka surat tajuk dan kulit tesis / disertasi.
(as it appears on the title page and front cover of the thesis / dissertation).

Bahawa tesis/disertasi tersebut boleh diterima dari segi bentuk serta kandungan dan meliputi bidang ilmu dengan memuaskan, sebagaimana yang ditunjukkan oleh calon dalam ujian lisan yang diadakan pada : *17 November 2015.*
That the said thesis/dissertation is acceptable in form and content and displays a satisfactory knowledge of the field of study as demonstrated by the candidate through an oral examination held on: November 17, 2015.

Pengerusi Viva: *(Chairman for VIVA)*	Prof. Dr. Che Su Mustaffa	Tandatangan *(Signature)*
Pemeriksa Luar: *(External Examiner)*	Prof. Dr. Siti Mariyam Hj Shamsuddin	Tandatangan *(Signature)*
Pemeriksa Dalam: *(Internal Examiner)*	Prof. Dr. Ku Ruhana Ku Mahamud	Tandatangan *(Signature)*
Nama Penyelia/Penyelia-penyelia: *(Name of Supervisor/Supervisors)*	Assoc. Prof. Dr. Azman Yasin	Tandatangan *(Signature)*
Nama Penyelia/Penyelia-penyelia: *(Name of Supervisor/Supervisors)*	Prof. Dr. Horizon Walker Gitano-Briggs	Tandatangan *(Signature)*

Tarikh:
(Date) *November 17, 2015*

Permission to Use

In presenting this thesis in fulfilment of the requirements for a postgraduate degree from Universiti Utara Malaysia, I agree that the Universiti Library may make it freely available for inspection. I further agree that permission for the copying of this thesis in any manner, in whole or in part, for scholarly purpose may be granted by my supervisor(s) or, in their absence, by the Dean of Awang Had Salleh Graduate School of Arts and Sciences. It is understood that any copying or publication or use of this thesis or parts thereof for financial gain shall not be allowed without my written permission. It is also understood that due recognition shall be given to me and to Universiti Utara Malaysia for any scholarly use which may be made of any material from my thesis.

Requests for permission to copy or to make other use of materials in this thesis, in whole or in part, should be addressed to:

Dean of Awang Had Salleh Graduate School of Arts and Sciences
UUM College of Arts and Sciences
Universiti Utara Malaysia
06010 UUM Sintok

Universiti Utara Malaysia

Abstrak

Dalam teori keputusan, Model Jumlah Wajaran (WSM) adalah kaedah terbaik dalam Analisa Keputusan Multi-Kriteria (MCDA) untuk menilai beberapa alternatif dari segi bilangan keputusan kriteria. Penetapan wajaran merupakan tugas yang sukar, terutama jika bilangan kriteria adalah besar dan kriteria tersebut mempunyai ciri yang berbeza. Terdapat beberapa masalah dalam dunia sebenar yang menggunakan kriteria yang bercanggah dan kesan bersama. Dalam bidang automotif, fenomena ketukan dalam enjin pembakaran atau pencucuhan bunga api dalaman menghadkan kecekapan enjin. Kuasa dan ekonomi bahan api boleh dimaksimumkan dengan mengoptimumkan beberapa faktor yang mempengaruhi fenomena ketukan, seperti suhu, sensor kedudukan pendikit, masa pencucuhan bunga api, dan revolusi per minit. Mengesan ketukan dan mengawal factor atau kriteria di atas membolehkan enjin berjalan pada kuasa dan bahan api terbaik ekonomi. Keputusan terbaik mesti diambil daripada trade-off yang paling optimum dalam pemilihan kriteria tersebut. Objektif utama kajian ini adalah untuk mencadangkan satu model baharu Fungsi Penilaian Aggregat Bukan-Wajaran (NWAEF) untuk bukan linear fungsi multi-objektif yang akan meniru tingkah laku ketukan enjin (pembolehubah bersandar bukan linear) untuk mengoptimumkan keputusan faktor bukan linear (pembolehubah bebas bukan linear). Kajian ini telah memberi tumpuan kepada pembinaan satu model NWAEF dengan menggunakan keluk teknik pemasangan dan derivatif separa. Ia juga bertujuan untuk mengoptimumkan sifat bukan linear satu faktor dengan menggunakan Algoritma Genetik (GA) dan juga menyiasat tingkah laku fungsi tersebut. Kajian ini mengandaikan bahawa pengaruh separa dan bersama antara faktor diperlukan sebelum faktor boleh dioptimumkan. The Kriteria Maklumat Akaike (AIC) digunakan untuk mengimbangi kerumitan model dan kehilangan data, yang boleh membantu menilai pelbagai model yang diuji dan memilih yang terbaik. Beberapa kaedah statistik juga digunakan dalam kajian ini untuk menilai dan mengenal pasti penjelasan yang lebih baik dalam model. Terbitan pertama digunakan untuk memudahkan bentuk fungsi penilaian. Model NWAEF telah dibandingkan dengan Genetik Algorithm Wajaran Rawak (RWGA) dengan menggunakan lima set data yang diambil daripada enjin pembakaran dalaman yang berbeza. Terdapat variasi yang agak besar di masa berlalu untuk mendapatkan penyelesaian terbaik antara kedua-dua model. Keputusan pengujian dalam keadaan sebenar (enjin pembakaran dalaman) menunjukkan bahawa model baharu mengambil bahagian dalam mengurangkan masa yang berlalu. Kajian ini merupakan bentuk kawalan ketukan dalam subruang yang boleh meningkatkan kecekapan dan prestasi enjin, meningkatkan ekonomi bahan api dan mengurangkan pelepasan terkawal dan pencemaran. Digabungkan dengan konsep baru dalam reka bentuk enjin, model ini boleh digunakan untuk meningkatkan strategi kawalan dan menyediakan maklumat yang tepat kepada Unit Kawalan Enjin (ECU), yang akan mengawal ketukan pantas dan memastikan keadaan engine yang sempurna.

Kata kunci: Model Jumlah Wajaran, Analisa Keputusan Multi-Criteria, Alogritma Genetik, Kriteria Maklumat Akaike, Keluk Pemasangan

Abstract

In decision theory, the weighted sum model (WSM) is the best known Multi-Criteria Decision Analysis (MCDA) approach for evaluating a number of alternatives in terms of a number of decision criteria. Assigning weights is a difficult task, especially if the number of criteria is large and the criteria are very different in character. There are some problems in the real world which utilize conflicting criteria and mutual effect. In the field of automotive, the knocking phenomenon in internal combustion or spark ignition engines limits the efficiency of the engine. Power and fuel economy can be maximized by optimizing some factors that affect the knocking phenomenon, such as temperature, throttle position sensor, spark ignition timing, and revolution per minute. Detecting knocks and controlling the above factors or criteria may allow the engine to run at the best power and fuel economy. The best decision must arise from selecting the optimum trade-off within the above criteria. The main objective of this study was to proposed a new Non-Weighted Aggregate Evaluation Function (NWAEF) model for non-linear multi-objectives function which will simulate the engine knock behavior (non-linear dependent variable) in order to optimize non-linear decision factors (non-linear independent variables). This study has focused on the construction of a NWAEF model by using a curve fitting technique and partial derivatives. It also aims to optimize the non-linear nature of the factors by using Genetic Algorithm (GA) as well as investigate the behavior of such function. This study assumes that a partial and mutual influence between factors is required before such factors can be optimized. The Akaike Information Criterion (AIC) is used to balance the complexity of the model and the data loss, which can help assess the range of the tested models and choose the best ones. Some statistical tools are also used in this thesis to assess and identify the most powerful explanation in the model. The first derivative is used to simplify the form of evaluation function. The NWAEF model was compared to Random Weights Genetic Algorithm (RWGA) model by using five data sets taken from different internal combustion engines. There was a relatively large variation in elapsed time to get to the best solution between the two models. Experimental results in application aspect (Internal combustion engines) show that the new model participates in decreasing the elapsed time. This research provides a form of knock control within the subspace that can enhance the efficiency and performance of the engine, improve fuel economy, and reduce regulated emissions and pollution. Combined with new concepts in the engine design, this model can be used for improving the control strategies and providing accurate information to the Engine Control Unit (ECU), which will control the knock faster and ensure the perfect condition of the engine.

Keywords: Weighted Sum Model, Multi-Criteria Decision Analysis, Genetic Algorithms, Akaike Information Criterion, Curve Fitting

Universiti Utara Malaysia

Acknowledgement

First of all I have to express my thanks and gratitude to Allah who gives me the ability to achieve this imperfect work and without his blessing and support nothing can be done.

It gives me great pleasure to express my gratefulness to everyone who contributed in completing this thesis. It was my pleasure to study under Associate Professor Dr. Azman Yasin's supervision. I'm so grateful for his support during the Years period of the study.

I would like to thank my co-supervisor Professor Dr. Horizon Gitano for his advanced ideas and his noble mind. His continuous advice and important comments helped improve my work successfully.

To my father, whose surname I proudly carry – I am forever appreciative. I hope he is proud of me even if he is no longer with us. To my mother, who gave me life and prayed for me all the time, may Allah continuously bless her with good health. To my brother, thanks for their love and support. To my wife Bayadir, who gave her time and patience during the years of study, I thank her from the deeps of my heart. I would also like to thank my two young babies Mustafa and Fatima, without whom my goal would not have been achieved. I dedicate this work to my family.

I thank all the workers in the UUM my university and school of computing to offer all support and facilities to complete this simple work.

UUM Universiti Utara Malaysia

Table of Contents

Universiti Utara Malaysia

Universiti Utara Malaysia

Universiti Utara Malaysia

Universiti Utara Malaysia

List of Tables

Universiti Utara Malaysia

List of Figures

Universiti Utara Malaysia

Universiti Utara Malaysia

List of Appendices

Universiti Utara Malaysia

List of Abbreviations

AIC	Akaik Information Criterion
ANOVA	Analysis of Variance
AEF	Aggregate Evaluation Function
AEFM	Aggregate Evaluation Function Model
AFPE	Akaike’s Final Prediction Error
AOFBPD	Aggregate of Objective Function-based Partial Derivative
CEP	Curve Expert Professional
CF	Curve Fitting
DOF	Degree Of Freedom
DM	Decision Making
ECU	Electronic Control Unit
EF	Evaluation Function
GAs	Genetic Algorithms
GCVC	Generalized Cross-Validation Criterion
GOF	Goodness Of Fit
IGN	Ignition Timing
kn	Knock
LSE	Least Square Error
ma	vector containing row numbers of mother chromosomes
MOOPs	Multi-Objective Optimization Problems
MOEF	Multi-objective Evaluation Function
MSE	Mean Square Error
N_{bits}	N_{gene} * N_{par}. : Number of bits in the chromosome
N_{var}	Number of variables
N_{gene}	Number of bits in the gene
N_{keep}	Number of chromosomes in the mating pool

Universiti Utara Malaysia

N_{pop}	Number of chromosomes in the population
NMOEF	Nonlinear Multi-objective Evaluation Function
NMOOPs	Nonlinear Multi-Objective Optimization Problems
pa	vector containing row numbers of father chromosomes
PD	partial Derivative
Rpm	Revolution Per Minute
RMS	Root Mean Square error or Standard error
SI	System Identification
SI	Spark Ignition
SNOPs	Single Nonlinear Objective Problems
SSE	Sum of Square Error
SST	Total Sum of Squares
Temp.	Temperature
Tps	Throttle position sensor
Varhi1	Highest number in the variable range
Varlo1	Lowest number in the variable range
VIF	Variance Inflation Factor
WSOF	Weighted Sum of Objective Functions
X_{rate}	Crossover rate

Universiti Utara Malaysia

CHAPTER ONE
INTRODUCTION

1.1 Background

Global optimization aims to find a solution for obtaining the global minimum (maximum) objective function. In other words, global optimization aims to determine not merely "a local minimum," but also "the smallest local minimum" with respect to the solution set. In the study of the problems of optimization, the focus is to look for optimal or near optimal solutions related to the goals stipulated (Rothlauf, 2011).

Problems in the sphere of global optimization refer to the optima of nonlinear functions being characterized and computed. These problems are common within the mathematical modelling of real systems and are found in a large array of applications. A huge number of theoretical, computational and algorithmic contributions have evolved over the past few decades, which have led to the solution of many global issues involving essential practical application.

When a non-linear relationship exists between entities, changes to one of those entities will not result in a change to the other entity. This means that the relationship that exists between the two entities can be considered unpredictable. Non-linear entities may possess relations that appear rather predictable but are more complex compared to linear relationships.

Optimization problems have been considered crucial because of their visibility and strength. All designs and engineering activities have multiple objectives because they are

Universiti Utara Malaysia

rooted in and are inherent of the four main objectives in the product and system design. Figure1.1 illustrates these objectives, which include performance, cost, scheduling (time), and risks (Maier & Rechtin, 2000).

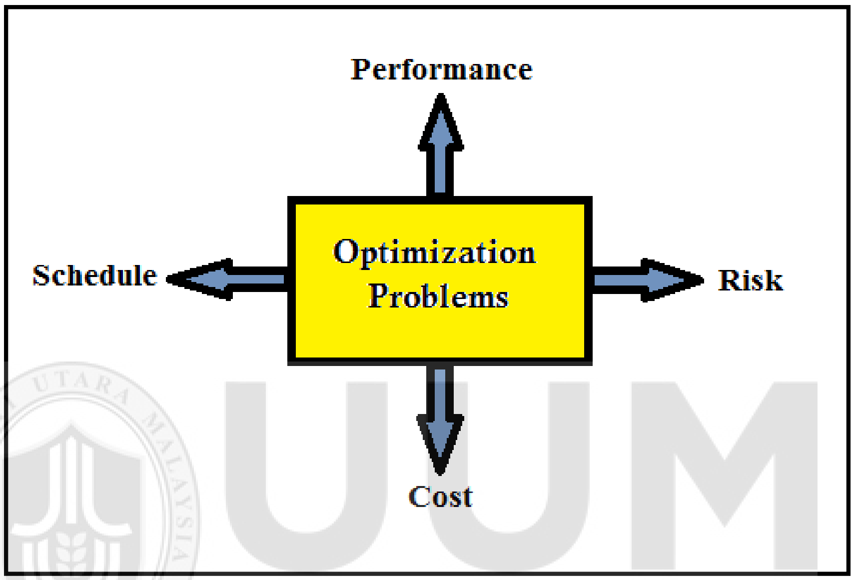


Figure 1.1. Factors of designs and engineering activities

Universiti Utara Malaysia

A mathematical model of the system must exist for each optimization problem. For example, in the field of automotives (Lochau, Sun, Goltz, & Huhn, 2010), the knocking phenomenon in internal combustion or spark ignition (SI) engines limits the efficiency of the engine(Eltaher, 2013; Lonari, 2011; Peyton, 2014). Abnormal combustion occurs at the time of an increase in pressure and temperature of non-burning gas contained within the cylinder, which will lead to the self-combustion of the fuel. This phenomenon leads to the combustion chamber experiencing oscillating pressure waves. The oscillating pressure is what causes damage which can shorten the life of the engine (Eltaher, 2013; Ganestam, 2010; Kasseris, 2011; Kozarac, Tomic, Taritas, Chen, & Dibble, 2015; Lonari, 2011; Peyton Jones, Spelina, & Frey, 2013; Taglialatela, Moselli, & Lavorgna, 2005; Thomasson et al., 2013). The knocking phenomenon is one of the major factors that limit the efficiency of SI engines (Lonari, 2011; Vancoillie, Sileghem, & Verhelst, 2013). These engines are being developed today to balance the reduction of fuel consumption with the improvement of the torque (Zhen et al., 2012). Control systems are designed in modern engines to minimize their exhaust emissions and to maximize their power and economy(Merola, Sementa, & Tornatore, 2011). Power and fuel economy can be maximized by optimizing some factors that affect the knocking phenomenon, such as temperature (TEMP), throttle position sensor (TPS), spark ignition timing(IGN), and revolution per minute (RPM) for a specific air/fuel ratio(Kozarac et al., 2015). Detecting knocks and controlling the above factors may allow the engine to run at the best power and fuel economy. Normal combustion occurs when a mixture of air and fuel is ignited using a spark plug; the combustion then flows smoothly from the point of ignition to the walls of the cylinder (Kasseris, 2011; Revier, 2006).

Universiti Utara Malaysia

Control of knock phenomenon is becoming more and more important in modern SI engine, due to the tendency to develop high boosted turbocharged engines (downsizing). To this aim, improved modelling and experimental techniques are required to precisely define the maximum allowable spark advance (Bozza, De Bellis, & Siano, 2014).

Given its excessive complexity, white-box modelling is no longer valuable for control applications. Therefore, simulating these plants in black-box form is required to develop a model by utilizing information from the system tests (Sarker & Newton, 2007; Vossoughi & Rezazadeh, 2004). Several design optimization problems can be modelled and solved as single nonlinear objective problems (SNOPs), which are usually highly constrained(Sarker & Newton, 2007).

A model that evaluates evolution must be built to achieve optimization. Such objective can be achieved through the “Evaluation Function” (EF). Evolution is not a purely random process, but must always be clarified or implied. EF serves as a guide for evolution. In other contexts, this function is termed objective function, fitness function, penalty function, profit function, cost function, scalar function, and energy function. EF assesses the status of the entire system (or one agent) and then selects the next step according to the value of the evaluation. Aside from having a key role, EF is also among the fundamental problems in evolution (Jing, 2005). Given that a single objective GA assesses the value for each individual solution in the population, evaluation tools must be used for the assessment. These tools must be accurate enough to generate an acceptable quantitative estimation of real world phenomena that are referred to by the optimization model (Ahmadi, 2007).

Universiti Utara Malaysia

Each issue is most likely to possess an objective, meaning the thing that one is trying to find. An objective refers to be the goals of a particular issue. The described goals are transferred to evaluation functions, which are capable of providing a map from the solution space to the number set (Michalewicz, Schmidt, Michalewicz, & Chiriac, 2005) (see Figure 1.2).

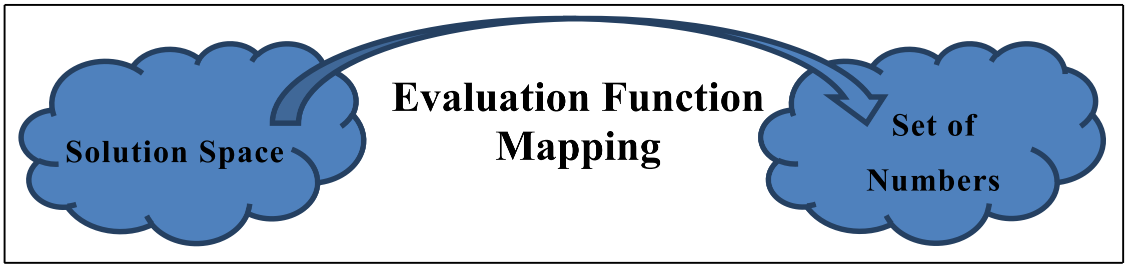


Figure 1.2. Mapping from the solution space to a set of numbers (Michalewicz et al., 2005)

Therefore, each solution is allocated a numeric value from the evaluation function for each specific goal. Because they usually articulate the connection that exists between the method and the problem(see Figure 1.3) (Michalewicz et al., 2005), evaluation functions (in the case of single objective) or a set of evaluation functions (in the case of multi-objective) are considered primary components of a heuristic method, regardless of whether it is tabu search, genetic algorithm, simulated annealing, ant system or simple hill climber (Michalewicz et al., 2005).

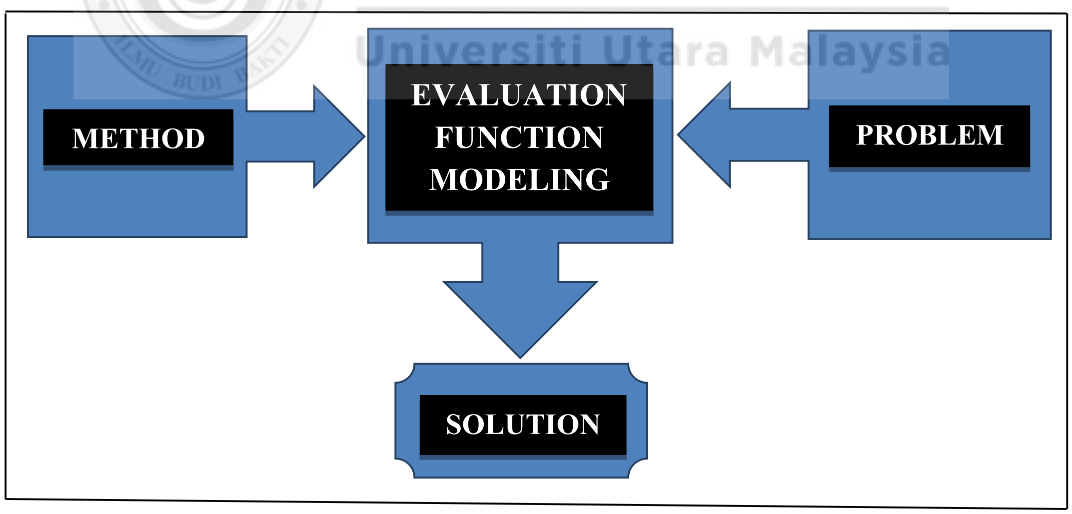


Figure 1.3. Connection between the method and the problem

Evaluation functions compare the quality of various candidate solutions by assigning each solution with a quality measure. The functions are capable of several things, including the return of a rank of the candidate solution in terms of a set of solutions, generating an exact number in the event of an evaluation function being defined as a closed formula, or may include several components, including penalty expression in the case where a candidate solution may violate problem specific constraints (Michalewicz et al., 2005).

The observation of MOOPs is considered the main advantage of GA over classic methods in optimization problems. The issues may be present in many scenarios, including product design, where certain criteria are required to be simultaneously satisfied (Shan & Wang, 2005; Tappeta, Renaud, & Rodríguez, 2002; Wilson, Cappelleri, Simpson, & Frecker, 2001), especially in cases in which many decision variables are present and the nature of the problem involves a complex trade-off. Within the context, 'trade-off' will refer to the situation where the value of one objective is traded for the value of another function (Ahmadi, 2007).

Universiti Utara Malaysia

There are some problems in the real world which utilise conflicting criteria and mutual effect. The best decision must arise from selecting the optimum trade-off within the criteria. Therefore, a new approach that is based on modelling and partial derivatives is proposed for the designing of a non-linear multi-objective evaluation function, which is the goal of nonlinear MOOPs.

1.2 Problem Statement

According to Grodzevich and Romanko (2006); Knowles and Hughes (2005); Sindhya (2011); Talbi, Basseur, Nebro, and Alba (2012), all engineering activities and designs mainly apply multi-objective problems. Many approaches have been applied for addressing MOOPs. These approaches can be classified into enumerative, numerical, and Guided random search techniques methods (Bandyopadhyay & Saha, 2012). Guided random search techniques methods have no assumptions or auxiliary information about the objective function.

One of the most important of nonlinear multi-objectives optimization problems in internal combustion engines (Spark ignition engine (SI)) is “knock”. Engine knock is an undesired phenomenon in spark ignited internal combustion engines, where it does not have a standard mathematical model to represent this phenomenon. Knock is still relevant and challenging (Peyton, 2014). There are many literature on knock sensing and detection, but needs significant potential in knock control. There are many nonlinear factors suffer from conflict and mutual influence that affect in knocking. A trade off for these factors is needed to prevent this problem (Kozarac et al., 2015; Millo, Rolando, Pautasso, & Servetto, 2014).

Universiti Utara Malaysia

Only guided random search techniques algorithms are capable of solving general nonlinear optimization issues with arbitrary objective functions utilising minimal quantities of assumptions regarding the objective function or by exempting limitation to (small) enumerative problems. The objective function weighting problem is a characteristic property of multi-objective problems (Herwijnen, 2011; Ismail & Yusof, 2010; Murata & Ishibuchi, 1995; Murata, Ishibuchi, & Tanaka, 1996; Tran, Hanif, Tölli, & Juntti, 2012). The solution for the weighting problem is a natural basis for the classification. The decision maker must make a decision regarding the relative importance of every objective function in order to obtain one unique solution for an original multidisciplinary decision-making problem. This decision can be performed by applying one of the three approaches (Ching-Lai & Abu, 1979; Marler & Arora, 2010; Zadbood & Noghondarian, 2012), firstly, in a priori preference articulation, the decision maker selects the weighting before running the optimization algorithm. Secondly, in progressive preference articulation, the decision maker interacts with the optimization program during the optimization process. Thirdly, in a posteriori preference articulation, no weighting is specified by the user before or during the optimization process.

The multi-objective optimization algorithm provides a set of efficient candidate solutions from which the decision maker may choose his/her solution. Two mainstream approaches are currently being used for defining MOOPs and conflicting objectives(Amouzgar., 2012; Jubril, 2012; Trummer & Koch, 2014; Yang, Karamanoglu, & He, 2013), namely, aggregate objective functions (weighted sum of objective functions) and pareto optimization(Hu & Mehrotra, 2012).

In aggregate multi-objective functions the following questions must be addressed, how conflicting (Santana-Quintero, Montano, & Coello, 2010; Yang, 2011) and mutually independent objectives are traded off (Ryu, Kim, & Wan, 2009); which objective must be favoured over the others; and how the individual objective functions must be weighted in relation to each other. Assigning weights is a difficult task, especially if the objectives are in large quantities and have distinct characteristics (Coello, 1999; Gabli, Jaara, & Mermri, 2014; Herwijnen, 2011; Ismail & Yusof, 2010; Kim & Weck, 2005; Konak, Coit, & Smith, 2006; Tran et al., 2012).

For this reason, a new Non-Weighted Aggregate Evaluation Function (NWAEF) for non-linear multi-objectives has been proposed. Evaluation function (EF) accuracy can be enhanced by selecting a suitable form for each decision variable (predictor variable) that has greatest effect on the dependent variable. The optimization accuracy is measured by answering the following research question:

1.3 Research Questions

1- How to select the factors (partial decision variables) which have most effect into decision-making problem?

2- How to construct non-weighted aggregate single nonlinear multi-objective evaluation function in order to optimize and mitigate engine knocking?

3- How to trade off the conflict and the mutual influence between individual objectives?

4- How to evaluate the non-weighted aggregate single nonlinear multi-objective evaluation function?

Universiti Utara Malaysia

1.4 Research Objectives

The main objective of this research is to propose a new Non-Weighted Aggregate Evaluation Function for Non-linear Multi-objectives (NWAEF) which will simulate the knock behavior (non-linear dependent variable) in order to optimize non-linear decision factors (non-linear independent variables). The following specific research objectives are fulfilled:

1- To construct and test regression models that having most influence using ANOVA and some statistical tests.

2- To identify the optimal nonlinear mathematical models for system identification modeling using curve fitting technique in order to optimizing, thus mitigate engine knocking.

3- To prevent conflicting and mutual effect by applying aggregate partial derivatives method for each objectives.

4- To evaluate the non-weighted aggregate single nonlinear multi-objective evaluation function using genetic algorithm (GA) in terms of accuracy.

1.5 Motivation and Significance of the Research

A generic tool-based framework for the automated application of a configurable GA was developed to a particular engine model. Overcoming weights through construction Non-Weighted Aggregate Evaluation Function (NWAEF) by using curve fitting, and partial derivative techniques, are an aid in providing a good approximation to feasible optimal solutions. In comparable with other models, some advantages can be obtained, namely:

- Overcome the non-linear weight selection implicitly greatly improved computational efficiency of the aggregate multi-objective method by reducing the complexity of model objectives.
- Time efficient (short time to solution), even when the problems have a large number of variables.
- Provides good approximations to feasible optimal solutions.

Universiti Utara Malaysia

- In building, a mathematical model we do not need information from the expert.
- Improved accuracy in solutions (reduce the error).
- Can add other objectives easily (expansion model).
- Further study of the relationships among the design variables in the model, which may allow the utilization of GAs on different models.

The number of electronically influenced parameters increases and is mainly controlled by the Engine Control Unit (ECU), setting up the ECU to find optimized engine parameterizations for specific operating points becomes more complicated (mutual influence and conflicting) . This study handled more factors although its complexity, it reached higher accuracy results from other models. In other hand, in internal combustion engine, the adjustment of purely mechanical control parameters is understood and handled accordingly. Along with the upcoming requirements some contributions may be provided including:

Universiti Utara Malaysia

- Reduction of fuel consumption and emissions. In addition preventing the engine damage, this research provides a form of knock control within the subspace that can enhance the efficiency and performance of the engine, improve fuel economy, and reduce regulated emissions and pollution.
- Using a real engine test bed to validate different settings for a stepwise approximation of optimal parameterization is also time-consuming and costly, therefore simulation has been used to build the model.
- Combined with new concepts in the engine design, this model can be used for improving the control strategies and providing accurate information to the ECU, which will control the knock faster and ensure the perfect condition of the engine.

- Provide the appropriate values (optimal values) for decision making, this model simulates a knock problem in SI engines by speeding up the decision that is taken by the ECU.

1.6 Scope, Assumption, and Limitations of the Research

This thesis has focused on the construction of a non-weighted aggregate evaluation function of multi-objective optimization for engine knock modelling by using a curve fitting technique and partial derivatives. This thesis also aims to optimize the non-linear nature of the factors by using GAs as well as investigate the behaviour of such function. This research assumes that a partial and mutual influence between factors is required before such factors can be optimized. The Akaik Information Criterion (AIC) is used to balance the complexity of the model and the data loss, which can help assess the range of the tested models and choose the best ones. Some statistical tools are also used in this thesis to assess and identify the most powerful explanation in the model. The first derivative is used to simplify the form of EF.

All tests have been performed to build an EF that is based on real data that are obtained by test engines in the Research and Development Center of the Malaysian Proton Company, as well as in those data that are obtained by the vehicle diagnostics tools of EGMA in Iraq. These data cover different situations and speeds (1000 rpm to 5000 rpm) to obtain a more reliable function.

The knocking phenomenon can be affected by many factors. Given the complexity on finding an appropriate analysis, this research only focuses on the three most significant of these factors.

1.7 Thesis Organization

This thesis consists of six chapters. Chapter one presents the research problem.

Chapter two is divided into three parts. The first part reviews the optimization, the second part deals with multi-objective optimization, and the third part reviews the multi-objective EF of the aggregation technique. The first part also covers the concepts, scope, methods, and problems of optimization as well as reviews related studies on such subject. The second part discusses the methods, problems, and previous studies that are related to MOOPs. The third part covers the concepts, problems, and related works on the multi-objective EF of the aggregation technique.

Chapter three reviews the methodology achieving the research objective. This chapter is divided into several subsections. The first subsection describes the general framework and the behaviour that has been examined in this thesis. The second subsection reviews how the datasets within the application can be obtained. The third subsection describes the selection of the best model for each objective. The fourth subsection describes the construction of the single multi-objective EF. The fifth subsection discusses the optimization of the single multi-objective EF.

Chapter four discusses the practical part of this thesis in more detail by reviewing the stages of building the EF and simulating the nonlinear factors that affect the knocking phenomenon.

Chapter five presents the evaluation results that are obtained by applying the EF to different conditions and vehicles to prove the effectiveness of the proposed model.

Universiti Utara Malaysia

Finally, Chapter six provides the conclusion of the whole research study, achievement of research, discussion the Knock detection methods, contributions, limitations and put forward some recommendations for future work.

Universiti Utara Malaysia

CHAPTER TWO
LITERATURE REVIEW

2.1 Introduction

This chapter, presents reviews of research works related to the field research study undertaken for this thesis. Section 2.1, this section provides an introduction to this chapter, 2.2 presents the concept of optimization and its techniques, while Section 2.3 introduces the problems related to the scope of optimization. While Section 2.4 presents Optimization Problems, reviews the research studies on the nonlinear factors and evaluative functions of optimization, Section 2.5 explains the basic concept of multi-objective optimization problems and, its techniques including a literature review of this concept. Meanwhile Section 2.6 presents the concepts and research studies on aggregate multi-objective optimization (AMOO). Section 2.7 reviews the research studies on the non-linear knock factors optimization and evaluation function. Finally, Section 2.8 provides a summary of this chapter.

2.2 Basic concepts of Optimization

Optimization is a common concept in many disciplines and various domains. In the study of the problems of optimization, the focus is to seek optimal or near optimal solutions related to the goals stipulated (Ahmad, 2012). Usually, the problems cannot be solved in one step. It requires a process with guide lines on problem solving. Often, the process of problem solving involves different steps which take place in a sequence. According to Rothlauf (2011) the common steps used include recognizing and defining problems, construction and solving of models, and evaluation and implementation of solutions.

The goal of optimization is to seek optimal solution, or near optimal solution with little calculated effort. The effort of optimization can be measured in term of time (calculation of time) and space (computer memory). There are many methods of optimization, such as modern heuristic. These methods attempt to strike a balance between the quality in solution and effort. This means that an increase in effort (time and space) often leads to an increase in quality of the solutions (Rothlauf, 2011).

2.3 Scope of optimization problems

In a practical task, optimization can be defined as a following stage, for instance, to determine the best solution for a given system or process, within certain constraints. The task comprises several elements. The first one is Objective Function (OF) that provides the numerical quantitative value for the measurement of performance. The value of this function is recorded either as minimum or maximum. It objective function (OF) can be in the form of cost, yield, profit or system.

Universiti Utara Malaysia

The second element is a Predictive model that describes the behavior of the system. In the optimization process, the problem is translated into a set of equations and inequalities with restrictions such as the limitations that affect the performance of the system.

The last element is Variables, found in the predictive model. Often, these variables undergo modifications to meet the restrictions. This can usually be done with the multiple instances of changing values, that result in a region determined by the subspace of these variables. In many of the problems found in the field of engineering problems, this subspace is described by a set of decision variables which can be interpreted as the degree of freedom in analyzing the process (Biegler, 2010).

2.4 Optimization Problems

It is necessary to have an insight of the systems and product design in order to achieve the desired performance. Therefore, it is essential to adopt an effective and systematic approach in the process of making decisions and improving performances. Such operations require optimization strategies in order to achieve the desired goals. However, many problems may be encountered during these processes. The following section addresses some of these problems.

2.4.1 Solution Process in optimization

Usually, users, companies and other organizations are unable to choose freely from all the available decisions alternatives (Rothlauf, 2011), despite the limitations that restrict the number of available alternatives. Generally, the restrictions are technical limitations, law or interpersonal relations between humans.

According to Rothlauf (2011) the most difficult step in the optimization of problem is recognition of problems. This is because users or institutions have to abandon the current way of doing business and accept other (and perhaps better) ways of doing things.

According to Arora, Huang, and Hsieh (1994) Several algorithms for discrete-integer-continuous optimization problems were developed, among them branch and bound method, penalty function approach, rounding–off, cutting plane, simulated annealing, genetic algorithms, neural networks, and Lagrangian relaxation methods. It is observed that some of the methods for discrete variable optimization use the structure of the problem to speed up the search for the discrete solution. This class of methods is not suitable for implementation into a general purpose application (Arora et al., 1994).

The researchers Liu & J. Chen clarified in (2011), the importance of the use of genetic algorithms in solving optimization problems through the application of genetic algorithm to optimize the function. He explained their ease of use in solving the problems of functions optimization, where the author explained the genetic algorithm provides a general framework which can solve nonlinear, multi-model and multi-objective optimization problems of complex systems, it does not depend on the specific areas of the problem belongs to, and has been widely used in function optimization, automatic control, image processing, machine learning and other technology.

Universiti Utara Malaysia

In (1997) the researcher Al-Duwaish introduced a new method for the control of nonlinear systems using genetic algorithms. The proposed method formulates the nonlinear controller design as an optimization problem and genetic algorithms (GA) are used in the optimization process. Researcher used a model as a fitness function that contains a single input (one factor) in order to get a single output (SISO), researcher used a genetic algorithm (GA) for minimize the square of the error between the input and output.

Farther more, In constructing models, the author (Rothlauf) posited in (2011) that there are two other relevant aspects in the building of the model namely availability of relevant data and evaluation of the resulting model. Also in constructing models, according to

Schneeweiss (2003), to build a model with an appropriate level of abstraction is a difficult task . Often, may start with a realistic model, but the problem cannot be solved. When this occurs, the realistic model is simplified to a form that can be solved by the optimization of existing methods. There is a basic trade-off between the capability of optimization methods to solve a model (tractability) and the similarity between the model and the underlying problem in the real world (validity).

Ahmad (2012) described solution technique called Best Candidates Method (BCM) for solving optimization problems; the goal is to minimize the computation time to get the optimal solution. In this study he applied the BCM to the linear assignment problems (LAP) that is one of the optimization problems in the Operation Research (OR). The author he say, either, find all available combinations in sequential or parallel solution manner then compare the results to get the optimal one, but its need a very large computation time for a large scale problems, or try to reach directly the optimal solution using different methods. But all available methods not always reach the optimal solution and have a complex solution technique or have a long computation time (Ahmad, 2012). The BCM based on election the best candidates and the alternative in each row and cover all columns with at least one candidate, then he can obtain the combinations that must be have no any intersect means, one candidate for each row and column. The BCM comparing to the Hungarian Method as shown from the solution steps can obtained the best combinations with less computation time and without complexity. Finally, this approach can be used for linear optimization problems only.

Universiti Utara Malaysia

In solving models, also Rothlauf (2011) believes that modern heuristics can be easily applied to problems that are very realistic and close to real-world matters. However, modern heuristics usually do not guarantee a perfect solution. However, the solution model is difficult as standard variables of modern heuristics, typically show limited performance. Only variables to a specific problem and specific model produced high-quality solutions (Bonissone, Subbu, Eklund, & Kiehl, 2006; Droste & Wiesmann, 2003; Puchta & Gottlieb, 2002).

In the previous sections that discussed optimization problems, one of most important solution process problem is the construction of the models. Many real-life problems are nonlinear. For example, if a person wants to buy a car, he/she will prefer both high performance and low cost.

Linear models are commonly used to approximate behavior. This innovative idea of using the linear framework is established well. Furthermore, linear models are easy to interpret and understand. Constructing linear models usually require less effort than nonlinear models. Unfortunately, linear approximations are only valid for a particular input (a given input range). Thus nonlinear modeling was preferred in different application areas in the past decades. Technological innovations have resulted in lesser restrictions in computation, memory, and access to data, making nonlinear modeling a more suitable option than linear modeling. According to Paduart et al. (2010), to build models for the nonlinear devices studied, employed method to identify the system. The primary goal of the system is to identify available mathematical models of input/output data. This goal is often achieved by minimizing cost function, which is an integral part of the statistical framework (Figure 2.1) (Paduart et al., 2010).

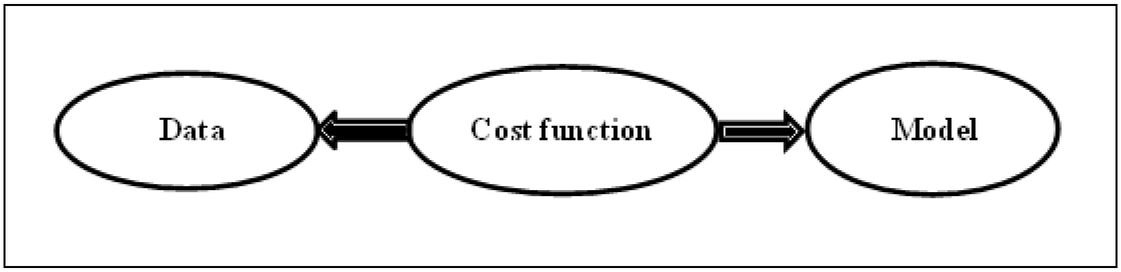


Figure 2.1. Basic idea of system identification: cost function relates data and model (Paduart et al., 2010)

2.4.2 Properties of Optimization Problems

The difficulty in the description of a problem lies with locating or identifying an optimal solution for a specific problem or problem instance. This difficulty in the identification of the problem is independent of the optimization method used. Determining the level of difficulty of a problem is often a challenging task as it is necessary to prove that there are no optimization methods that can better solve the problem. Therefore, statements about the levels of difficulty of a problem are method independent as they must hold for all possible optimization methods.

It is known that different types of optimization methods lead to different search performances. Often, optimization methods exploit the characteristics of an optimized problem show better performance. In contrast, methods that do not exploit any characteristics of the optimized problem such as black-box optimization techniques, usually show low performance. It is imperative to look into random search to better understand its application.

According to Rothlauf (2011), there are many properties that related to optimization problems. these properties like problem difficulty, locality, decomposability. Locality of a problem is generally a description of how well the distances $d\ (x,y)$ between any two solutions $x,y \in X$ correspond to the difference in the objective values $|f(x) - f(y)|$ (Franz, 2006; Lohmann, 1993; Rechenberg, 1994). The locality of a problem is high if neighboring solutions have similar objective values and is low if the distances do not correspond to the differences in objective values. Relevant determinants for the locality of a problem are the metrics defined in the search space and the objective function f.

In heuristic literature, several studies focus on locality for discrete decision variables (Caminiti & Petreschi, 2005; Gottlieb, Julstrom, Raidl, & Rothlauf, 2001; Gottlieb & Raidl, 2000; Paulden & Smith, 2006; Raidl & Gottlieb, 2005; Rothlauf & Goldberg, 1999; Weicker & Weicker, 1999; Whitley & Rowe, 2005) and for continuous decision variables (Igel, 1998; Rechenberg, 1994; Sendhoff, Kreutz, & Von Seelen, 1997). For continuous decision variables, locality is also known as causality. High and low localities correspond to strong and weak causalities respectively.

2.4.3 Reviewing of Optimization Methods

Two different types of optimization methods can be identified. ***Exact optimization methods*** can guarantee that an optimal solution will be found, whereas ***heuristic***

Universiti Utara Malaysia

optimization methods cannot. When the effort increases polynomially with problem size, exact optimization methods are preferred (Rothlauf, 2011).

Some medium-sized problems require exponential methods because the problems are intractable and cannot be solved using exact methods. Heuristic optimization methods are used to overcome these problems. These optimization methods are problem specific because they exploit the properties of the problem. The methods are also suitable for practical problems. Table 2.1 shows the specific strategies of the two optimization methods (R. Horst & H. E. Romeijn, 2002).

Table 2.1

Optimization Methods (Horst & Romeijn, 2002)

No.	EXACT METHODS	No.	HEURISTIC METHODS
1.	Adaptive stochastic search methods	1.	Approximate convex underestimation
2.	Bayesian search algorithms	2.	Continuation methods
3.	Branch-and-bound algorithms	3.	Genetic algorithms, evolution strategies
4.	Enumerative strategies	4.	“Globalized” extensions of local search methods
5.	Homotopy and trajectory methods	5.	Sequential improvement of local optima
6.	Integral methods	6.	Simulated annealing
7.	“Naive” (passive) approaches	7.	Tabu search (TS)
8.	Relaxation (outer approximation) strategies		

Universiti Utara Malaysia

2.4.3.1 Exact Methods

In (2002), the authors Horst & Romeijn illustrated ***Adaptive stochastic search methods***, these procedures depend, at least in part, on the random sample taken, adaptive search strategy adjustments, statistical stopping rules, clustering samples, and deterministic solution refinement options that can be entered as improvements on the original sample. These methods can be applied to Global Optimization Problems (GOPs) of discrete and continuous values with very general conditions (Horst & Romeijn, 2002). In other side of algorithms, ***Bayesian search algorithms***, these algorithms are based on some prior assumptions of a random model with the same class of function in that given instance. Subsequent adjustments for the estimate is based on this model and the actual results. Bayesian search algorithms are frequently used methods (single-step optimization), and thus accounting is easier. Approximate decisions govern the research procedures. Bayesian methods are applied to general Continuous Genetic Optimization Problems (CGOPs) (Mockus, 2010; Mockus, Eddy, Mockus, Mockus, & Reklaitis, 1996). Also, another procedures, called ***Branch-and-bound algorithms***, these procedures depend on adaptive partition, sampling, and bound (in partial subsets allowed in the main set). They can be applied to models with continuous GOs or mixed problems, analogous to pure integer programming, or mixed integer linear programming methodology. This general procedure applies to many special and specific situations and provides extensive generalizations. Branch-and-bound methods can be systematically applied to the various GOPs, such as concave minimization, DC programming, and Lipschitz optimization problems (Floudas, 1999; Hansen, 1992; Horst & Tuy, 1996a; Kearfott, 1996; Neumaier, 1990; Pintér, 1996a; Ratschek & Rokne, 1988; Strongin & Sergeyev, 2000).

According to Horst and Tuy (1996b), another methods are used, like ***Enumerative strategies,*** these methods rely on the enumeration of all possible solutions. They are applicable to combinatorial optimization problems and to certain structured CGOP models (e.g., concave programming) (Horst & Tuy, 1996b). As well as, there is ***Homotopy and trajectory methods***, this strategy aims to visit all the fixed points of the objective function f in the main dataset and list all the global and local optimal points. The methodology can be applied to smooth GOPs, but mathematical analysis may be too exhaustive (Diener, 1995). In other side, there are ways are used like ***Integral methods***, these methods are designed to determine the main supremum of the objective function f of master data D by approximating the level sets of the function f (Hichert, Hoffmann, & Phú, 1997; Zheng & Zhuang, 1995). Also, other methods are used, called ***"Naive" (passive) approaches*** this approach includes simultaneous and random research on the grid. No correlation occurs between the sample points selected (samples can be taken simultaneously) without considering the individual results. Although such methods are obviously convergent under mild analytical assumptions, they are inappropriate in solving higher (often already in 3, 4, 5, etc.) dimensional problems (Pintér, 1996b; Zhigljavsky & Pintér, 1991). Finally, one of the exact methods, is ***Relaxation (outer approximation) strategies***, in this strategy, the general GOP is replaced with another sequence of partial problems (relaxed subproblems) that is difficult to solve. Cutting specific and general plans and various minorant functions are possible options. Relaxation algorithms are applicable to diversely structured GOs, such as concave minimization or DC programming models (Benson, 1995; Horst & Tuy, 1996b).

Universiti Utara Malaysia

2.4.3.2 Heuristic Methods

In (1997), the authors Dill et al. illustrated ***approximate convex underestimation***, this strategy attempts to determine the convexity properties of the objective function based on the main sample data in D. This method is effective in a number of situations; however, in other cases (i.e., the quadratic model is inappropriate) it will not produce reliable approximate solutions. These strategies can be applied to smooth GOPs (Dill et al., 1997). While, in (1997) the authors More & Wu clarified ***Continuation methods***, for these methods, the objective function is made simple and smooth with few local minimizers. Next, the minimization procedure is employed to trace all the minimizers and returned to the original function. These methods are used in smooth GOPs (Moré & Wu, 1997). In other side, ***Genetic algorithms (GA)*** are used, under ***evolution strategies***. These Evolutionary optimization methods mimic biological evolution models. Different types of specific algorithms that are deterministic, random, and rule-based can be built. These strategies can be applied to discrete and continuous GOPs under moderate structural requirements (Glover & Laguna, 1997; Michalewicz, 1996; Osman & Kelly, 1996; Voss, Osman, & Roucairol, 1999). Also, other methods are used, called **"Globalized" extensions of local search methods**, these practical strategies begin with a global search (random search) followed by a local search. These methods are applied to smooth the GOPs, and the differential is usually assumed to include the local search component (Pintér, 1996a; Zhigljavsky & Pintér, 1991). As well as, there is ***Sequential improvement of local optima***, these methods include tunneling, deflation, and filled function methods. These strategies run on adaptively constructed auxiliary functions to assist the search for the progressive arrival of optima while avoiding the ones found thus far. They are applicable to smooth GO problems (Levy & Gómez, 1985). As well as, is used ***Simulated annealing***, these techniques are based on the physical measurement of the cooling crystal structures that automatically link to a configuration with minimally stable energy, globally or locally. Simulated annealing is applicable to both discrete and continuous GOPs under mild structural requirements (Glover & Laguna, 1997; Osman & Kelly, 1996). ***Tabu search (TS),*** is used, the basic concept of this method is to prevent the movement of research to previously visited points (usually discrete) in the search space, at least within the next few steps. TS methodology is primarily used to solve combinatorial optimization problems, although it can also be extended to handle continuous GOPs (Glover & Laguna, 1997; Osman & Kelly, 1996; Voss et al., 1999).

Colby (2013) used the *difference evaluation function* to determine agent-specific feedback. This function has excellent empirical results in various domains, including air traffic control and mobile robot control. To transfer the computer constraint satisfaction problem (CSP) to a multi-agent system, in (2003), Jing & Qingsheng ("Emergence from Local Evaluation Function") describes several evaluation function algorithms that illustrate the relationship between the problems of computers and complex systems. Of these algorithms, local search (LEF) and simulated annealing in a multi-agent framework explain the traditional algorithms used in a global evaluation function (GEF) that computes how good the current system state is and determines the total violated constraints. The researcher also explained some algorithms (EO and Alife) that are used in LEF to self-organize to a global solution state. LEF determines how good the agent state is and the total violated constraints.

Universiti Utara Malaysia

In general, numerous methods are used in optimization to formulate and solve decision-making problems. In Figure 2.2, and according to Talbi (2009), optimization methods can be classified as follows.

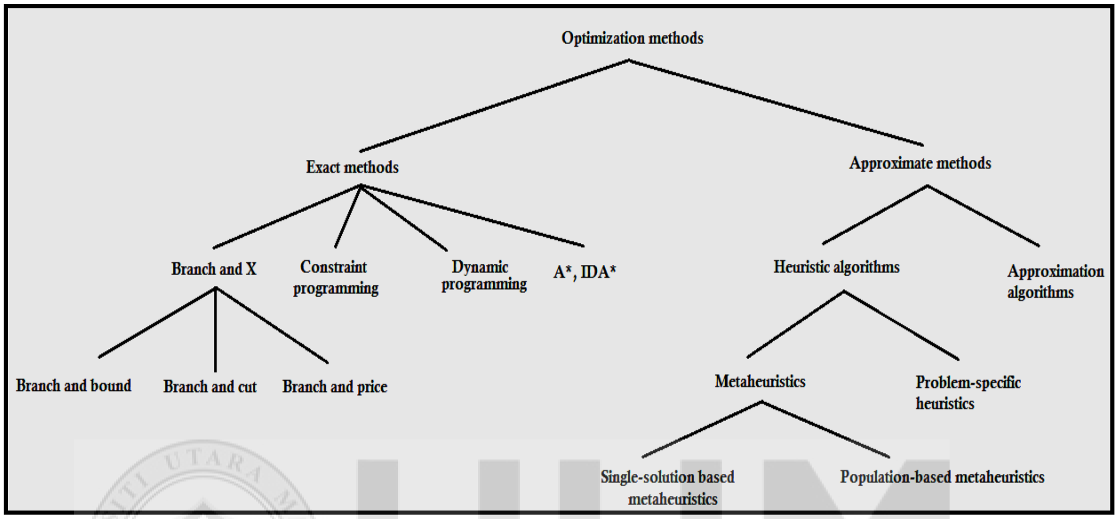


Figure 2.2. Classical optimization methods

Universiti Utara Malaysia

2.5 Multi-Objective optimization

Consciously or unconsciously, decisions are made every day of our lives. These decisions can be as simple as selecting a design of a dress or deciding the menu for dinner, or as difficult as designing a bridge or selecting a career. The former decisions are easy to make, whereas the latter may take several years because of the level of complexity involved. The main goal of most decisions is to optimize one or more criteria to achieve the desired result. Therefore, the development of optimization algorithms has been a great challenge in computer science (Bandyopadhyay & Saha, 2013). The problem is compounded by the fact that, in many situations, several objectives must be optimized simultaneously. These specific problems are known as multi-objective optimization problems (MOOPs). An array of metaheuristic single-objective optimization techniques, such as genetic algorithms, simulated annealing, differential evolution, and their multi-objective versions, have been developed.

2.5.1 Basic Concept of Multi-Objective Optimization Problems

At present, research on multi-objective optimization is very active because most real-world engineering optimization problems are multi-objective in nature (Coello, 2001; Luna, Nebro, & Alba, 2006; Zhou et al., 2011). The task of finding solutions for such problems is known as multi-objective optimization (MOO) (also called multi-criteria optimization). MOOPs have several objective functions to be optimized which are usually in conflict with on another(Zhou et al., 2011). Specifically, multi-objective optimization is a type of problem with solutions that can be evaluated along two or more incomparable or conflicting objectives. The general form of MOOP is described below.

Minimize/Maximize $f_m(x)$, $\quad$ m=1, 2,…..,M

Subject to

$g_j(x) >= 0$ $\quad$ j=1, 2, ….., J

$h_k(x) = 0$, $\quad$ k=1, 2,……,K

$x_i^L \leq x_i \leq x_i^U$ $\quad$ i=1, 2,…....,n

A solution x is a vector of n decision variables: x = x_1 , x_2,……., x_n

The last set of constraints is called variable bounds; these bounds restrict each decision variable x_i to take a value within a lower x_i^L and an upper x_i^U bound. They constitute a decision variable space D or the decision space.

MOO does not restrict the determination of a unique single solution, but does restrict a set of solutions collectively known as the Pareto front. Evolutionary algorithms (EAs) are suitable for solving such kinds of problems because they are capable of finding multiple trade-off solutions in a single run. Recognized subclasses of EAs include genetic algorithms (GA), genetic programming (GP), evolutionary programming (EP), and evolution strategies (ES) (Luna et al., 2006). Evolutionary algorithms are suitable in solving multi-objective optimization problems because such algorithms simultaneously deal with a set of possible solutions (population). Several members of the Pareto optimal set in a single run of the algorithm can be determined instead of performing a series of separate runs, as in the case of traditional mathematical programming techniques (Coello, 1999).

The first implementation of an Evolutionary Multi-Objective Optimization (EMOO) approach was the vector evaluation genetic algorithm (VEGA) Schaffer (1985) introduced in the mid-1980s and was mainly intended to solve problems in machine learning (Coello, 2001). However, these algorithms may be computationally expensive because (1) real-world problem optimization typically involves tasks demanding high computational resources, and (2) they aim to find the whole front of optimal solutions instead of searching for a single optimum (Luna et al., 2006). MOO does not restrict the determination of a unique single solution but does restrict a set of solutions called non-dominated solutions. Each solution in this set is said to be a Pareto optimum and is known as the Pareto font when they are plotted in the objective. Obtaining the Pareto front of a given MOP is the main goal of MOO. EMOO approaches are classified using the simple classification in Figure 2.3.

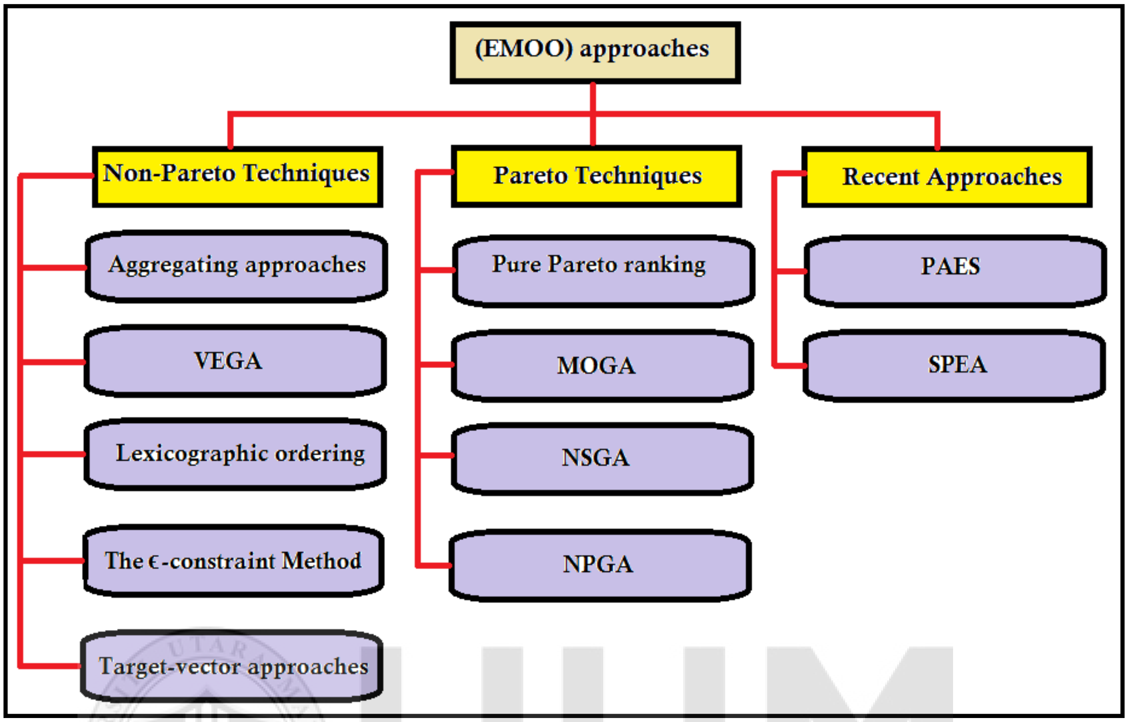


Figure: 2.3. Summarized approaches of EMOO (Coello, 2001)

UUM Universiti Utara Malaysia

2.5.2 Reviewing Evolutionary Multi-Objective Optimization Approaches

Various strategies address the problems of MOO. In this section, we will discuss the most common approaches used in such problems and clarify some of the advantages and disadvantages of each approach.

2.5.2.1 Aggregating Function Method

Genetic algorithms (GA) depend on a scalar fitness function to guide the search. The most intuitive approach to deal with multiple objectives is to combine these objectives into a single function. The approach of combining objectives into a single (scalar) function is normally a denominated aggregating function, which has been attempted several times in literature with relative success in problems where the behavior of the objective functions is more or less determined. An example of this approach is a sum of weights of the form:

$$\min \boldsymbol{f}(\mathbf{x}) = w_1 f_1'(\mathbf{x}) + w_2 f_2'(\mathbf{x}) + \ldots + w_k f_k'(\mathbf{x}) \quad (2.5)$$

where $f_i'(\mathbf{x})$ is the normalized objective function, $f_i(\mathbf{x})$ and $\Sigma w_i = 1$. This approach is called a priori approach because the user is expected to provide the weights. Solving a problem with the objective function (2.5) for a given weight vector $\mathbf{w} = [w_1, w_2, \ldots, w_k]$ yields a single solution, and if multiple solutions are desired, the problem should be solved multiple times with different weight combinations. To solve a MOO problem, weight w_i is assigned to each normalized objective function $f_i'(\mathbf{x})$, such that the problem is converted to a single objective problem with a scalar objective function as mentioned earlier (2.5) (Santana-Quintero et al., 2010).

This approach has several advantages. It is very simple, easy to implement, and efficient because it does not require any changes to the basic mechanism of a genetic algorithm. The approach can work properly in MOO problems with few objective functions and convex search spaces. However, one obvious problem of this approach is that generating a set of weights that properly scales the objectives when little is known about the problem may be difficult. However, the most serious drawback is that it cannot generate proper members of the Pareto optimal set when the Pareto front is concave, regardless of the weights used (Coello, 2001; Das & Dennis, 1997).

2.5.2.2 Vector Evaluated Genetic Algorithm

Schaffer (1985) proposed an approach called VEGA, which differs from GA in the way the selection is performed. This operator is modified so that at each generation, a number of sub-populations is generated by performing proportional selection according to each objective function. Thus, for a problem with k objectives and a population size of M, k sub-populations of size M/k each will be generated. These sub-populations will be shuffled together to obtain a new population of size M, on which the GA will apply the crossover and mutation operators in the usual way.

Advantages and Disadvantages. Given that only the selection mechanism of the GA needs to be modified, the approach is thus easy to implement and quite efficient. However, the "middling" problem prevents the technique from finding the compromise solutions that we normally aim to produce. In fact, if proportional selection is used with VEGA (as Schaffer did), the shuffling and merging of all the sub-populations will correspond to the average fitness components associated with each objective (Richardson, Palmer, Liepins, & Hilliard, 1989). Under these conditions, VEGA behaves as an aggregating approach subject to the same problems of such techniques.

Universiti Utara Malaysia

2.5.2.3 Multi-Objective Genetic Algorithm

Fonseca and Fleming (1993) proposed the *multi-objective genetic algorithm* (MOGA). The approach consists of a scheme where the rank of a certain individual corresponds to the number of individuals in the current population by which it is dominated. All non-dominated individuals are assigned rank 1, whereas the dominated ones are penalized according to the population density of the corresponding region of the trade-off surface (Coello, 2001).

Advantages and Disadvantages. The main strengths of MOGA is that it is efficient and relatively easy to implement (Coello, 1996). However, as with all the other Pareto ranking techniques, the performance of MOGA is highly dependent on the appropriate selection of the sharing factor.

MOGA is a very popular EMOO technique (particularly within the control community), and it normally exhibits very good overall performance (Coello, 1996).

2.5.2.4 Non-dominated Sorting Genetic Algorithm

Non-dominated sorting genetic algorithm (NSGA) was proposed by Srinivas and Deb (1994) based on several layers of individual classifications. Before selection (stochastic remainder proportionate selection is used), the population is ranked on the basis of domination (using Pareto ranking), and all non-dominated individuals are classified into one category (with a dummy fitness value that is proportional to the population size).

Advantages and Disadvantages. Some researchers reported that NSGA has lower overall performance than MOGA (both computationally and in the quality of the Pareto fronts produced) and seems to be more sensitive to the value of the sharing factor than MOGA (Coello, 1996). However, Deb et al. (2000) recently proposed a new version of this algorithm called NSGA-II, which is more efficient (computationally) and uses elitism and a crowded comparison operator that keeps diversity without specifying any additional parameters. The new approach has yet to be tested extensively, but it is certainly promising.

2.5.2.5 Niched Pareto Genetic Algorithm

Horn et al. (1994) proposed the Niched Pareto Genetic Algorithm (NPGA), which uses a tournament selection scheme based on Pareto dominance. Instead of limiting the comparison to two individuals (as normally done with traditional GAs), a higher number of individuals is involved in the competition (typically around 10% of the population size). When both competitors are either dominated or non-dominated (i.e., when there is a tie), the tournament result is decided through fitness sharing in the objective domain (a technique called equivalent class sharing is used in this case) (Horn et al., 1994).

Advantages and Disadvantages. This approach does not apply Pareto ranking to the entire population but only to a segment of it at each run; hence, its main strength is that it is faster than MOGA and NSGA4. NPGA also produces good non-dominated fronts that can be kept for a large number of generations (Coello, 1996). However, aside from requiring a sharing factor, this approach also needs an additional parameter that is the size of the tournament.

2.5.2.6 Target Vector Approaches

Approaches wherein the decision maker has to assign targets or goals for each objective are considered in this approach. In this case, the GA tries to minimize the difference between the current solution found and the vector of goals (different metrics can be used for that purpose). The most popular techniques are hybrids with goal programming (Deb, 1999; Wienke, Lucasius, & Kateman, 1992), goal attainment (Wilson & Macleod, 1993; Zebulum, Pacheco, & Vellasco, 1998), and the min-max approach (Coello & Christiansen, 1998; Hajela & Lin, 1992).

Advantages and Disadvantages. The main strength of these methods is their efficiency (computationally) because they do not require a Pareto ranking procedure. Their main weakness is the definition of the desired goals, which requires extra computational effort (normally, these goals are the optimum of each objective function considered separately). Furthermore, these techniques will yield a non-dominated solution only if the goals are chosen in the feasible domain, and such condition may certainly limit their applicability.

2.5.3 Review in Multi-Objective Genetic Algorithm

In (2007), Ahmadi employed single and multi-objective GA to find optimal solution(s) for design parameters of intake and exhaust systems, including intake pipes length, intake manifold geometry, timing of intake and exhaust valves, and exhaust manifold geometry. A model was used as an evaluation tool and genetic algorithm as an evolution method.

Optimization problem was solved in two cases. In the first case, MOGA was implemented to solve a multi-objective problem, and a single GA was developed for the second problem. As a result, an optimal design layout for intake and exhaust systems was chosen from Pareto-optimal solutions. For the second case, the best timing for intake and exhaust valve was found at each engine speed to aid in developing a variable valve timing system for the engine.

Universiti Utara Malaysia

An improved MOGA was proposed Liu et al. (2014) to solve constrained optimization problems. The constrained optimization problem was converted into a multi-objective optimization problem. The researcher presented an algorithm based on the multi-objective technique, where the population is divided into dominated and non-dominated subpopulations. An arithmetic crossover operator was utilized for the randomly selected individuals from dominated and non-dominated subpopulation. The crossover operator could gradually lead the individuals to the extreme point and improve the local searching ability. Diversity mutation operator was introduced for the non-dominated subpopulation.

In most response surface method (RSM) problems, the exact relationship between the response variables and the independent variables is not known. In 2012, Zadbood and Noghondarian (2012) dealt with multiple response surface (MRS) optimization problems with conflicting responses. They meticulously studied the most prominent approaches to MRS optimization, and reviewed and discussed the classifications of these approaches, with a special focus on the decision maker's preference information. They recommended three steps to solve these problems: collecting data, building a model, and optimization. A low-order polynomial was used to build a model for the relationship among the variables; for any curvature in the system, a polynomial of higher degrees, mostly second order, is applied (Zadbood & Noghondarian, 2012). Thus, they used a quadratic polynomial. Results of their case study showed that applying an interactive method with an existing MRS approach generates better results.

Jaszkiewicz, Hapke, and Kominek (2001) presented a comparative experiment with four multiple-objective evolutionary algorithms on a real-life combinatorial optimization

Universiti Utara Malaysia

problem. The test problem corresponds to the design of a distribution system. The experiment compares the performances of a multiple-objective multiple-start local search (MOMSLS), Pareto ranking-based multiple-objective genetic algorithm (Pareto GA), an extension of Pareto GA involving local search (Pareto GLS), and multiple-objective genetic local search (MOGLS). The results of their experiment clearly indicated that the method-hybridizing recombination and local search operators by far outperform methods that use only one of the operators. Furthermore, MOGLS outperformed Pareto GLS.

Zou, Liu, Kang, and He (2004) proposed a high-performance multi-objective evolutionary algorithm (HPMOEA) based on the principles of the minimal free energy in thermodynamics. The innovations of HPMOEA include the following: providing a new fitness assignment strategy by combining Pareto dominance relation and Gibbs entropy, and providing a new criterion for the selection of new individuals to maintain population diversity. They compared the performance of HPMOEA and those of four other well-known multi-objective evolutionary algorithms (MOEAs), namely, NSGA II, SPEA, PAES, and TDGA, on a number of test problems. Simulation results showed that the HPMOEA can find a much better spread of solutions and has better convergence near the true Pareto-optimal front on most problems.

A memetic algorithm designed by Adra, Griffin, and Fleming (2009) addressed the requirement for solution convergence toward the Pareto front of a multi-objective optimization problem. It incorporated a convergence accelerator operator (CAO) in existing algorithms for evolutionary multi-objective optimization. The convergence accelerator works by suggesting improved solutions in objective space and using neural

Universiti Utara Malaysia

network mapping schemes to predict the corresponding solution points in a decision variable space. The researchers used a number of objectives from two to eight. In all cases, the introduction of the CAO led to improved convergence for comparable numbers of function evaluations.

A new multi-objective optimization algorithm based on the emulation of the immune system behavior was proposed and validated by (Freschi & Repetto, 2005). The proposed approach was compared with the NSGA2 algorithm, which is representative of state-of-the-art approaches to multi-objective optimization. They used three standard problems (unconstrained and constrained) to test the algorithm and compare it with the NSGA2 algorithm. Three different metrics were adopted to carry out the comparisons. The authors claimed that the proposed approach can be a valid alternative to standard algorithms based on the results it obtained and its performance, which is similar or better than that of NSGA2.

Universiti Utara Malaysia

A real-life electromagnetic MOOP was dealt with by (Dias & De Vasconcelos, 2002). The authors proposed and described an NSGA, which they compared with four other algorithms (i.e., VEGA, NPGA, MOGA, and the classical method of objective weighting) using two test problems. From the comparison, the proposed NSGA performed better than the others, showing that it can be successfully used to find multiple Pareto-optimal solutions.

In the automotive field, the rising demands for better performance and fuel economy by consumers on the one hand and the very restrict emission standards on the other have forced the automotive industry to take prudent measures to step up to these challenges.

To this end, the role of vehicle engines, specifically their calibration, is crucial. Vossoughi and Rezazadeh (2005) proposed a multi-objective structure for the optimization of engine control unit (ECU) mapping. Two different MOGAs, namely, distance-based Pareto genetic algorithm and NSGA (together with entropy-based MOGA), were proposed and applied. The results demonstrated that the computerized structure was superior to the manual mapping methods, and that multi-objective methods have more generality compared with single-objective ones.

The knocking problem in an internal combustion engine belongs to a class of nonlinear problems, in which both steady-state and dynamic behaviors are nonlinear. Knocking control process requires intelligent monitoring because of the nonlinear nature of the knock and the nonlinear functional relationship between the input and output variables involved. In spite of continuing advances in optimal solution techniques for optimization and control problems, many of such problems remain too complex to be solved by known techniques. In mechanical and chemical engineering, evolutionary optimization has been applied by authors to identification systems (Dao, 2010; Pham & Coulter, 1995).

2.6 Aggregating Multi-Objective Optimization

Aggregating multi-objective optimization (AMOO) is the first technique developed to generate non-inferior solutions for multi-objective optimization. This technique is an obvious consequence of the seminal work of Kuhn and Tucker on numerical optimization (Coello, 1999; Kuhn & Tucker, 1951). This technique is also called "aggregating functions" because it combines (or "aggregates") all the objectives into a single objective. Addition, multiplication, or any other combination of mathematical operations can be used (Coello Coello, 2001). Considered the oldest mathematical programming method, aggregating functions can be derived from the Kuhn–Tucker conditions for non-dominated solutions. This technique type, however, does not incorporate directly the concept of Pareto optimum and is incapable of producing certain portions of the Pareto front. Characteristics of this technique are efficiency and ease of implementation, but it can handle only a few objectives (Coello, 2001).

2.6.1 Basic Concept of Aggregating Multi-Objective Optimization

Depending on how optimization and the decision process are combined, multi-objective optimization methods can be broadly classified into three categories (Coello, 2000; Horn, 1997; Hwang & Masud, 1979; Zadbood & Noghondarian, 2012).

Decision making before search: The objectives of the MOP are aggregated into a single objective, which implicitly includes preference information given by the decision maker (DM). In a priori preference articulation (Decide → Search), the DM combines the different objectives into a scalar cost function to effectively transform the MOP into a single-objective problem prior to optimization (Zadbood & Noghondarian, 2012).

Decision making during search: The DM can articulate preferences during the interactive optimization process. After each optimization step, a number of alternative trade-offs are presented based on the DM-specified further preference information, which respectively guides the search. In this technique, progressive preference articulation (Search ↔ Decide) decision making and optimization are intertwined. Partial preference information upon which optimization occurs is provided, providing an "updated" set of solutions for the DM to consider (Zadbood & Noghondarian, 2012).

Search before decision making: Optimization is performed without any preference information given. The result of the search process is a set of candidate (ideally Pareto optimal) solutions from which the final choice is made by the DM. This result means that a posteriori preference articulation (Search → Decide) DM is presented with a set of efficient candidate solutions and chooses from that set (Zadbood & Noghondarian, 2012).

Various methods fall under each MOEA solution technique, as summarized in Figure 2.4 (Coello, Lamont, & Van Veldhuizen, 2007).

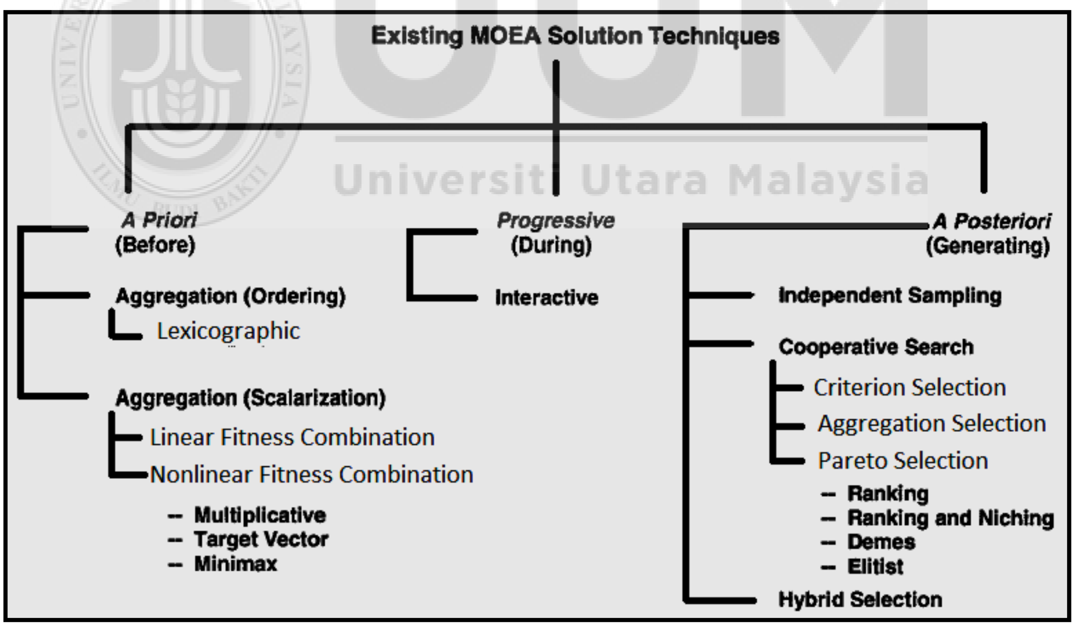


Figure 2.4. MOEA Solution Technique Classification (Coello et al., 2007).

The aggregation of multiple objectives into one optimization criterion has the advantage that the classical single-objective optimization strategies can be applied without further modifications (Zitzler, 1999).

In the aggregation method, a group of individual objectives is converted into one single objective using addition, multiplication, or other mathematical operations. One such method is weighting method or weighted sum, through which the original MOP is converted into a single-objective problem by forming a linear combination of the objectives, such as the following:

$Y = f(x)= w_1.f_1(x) + w_2.f_2(x) +\ldots\ldots+ w_k.f_k(x),$

The weights are denoted by w_i and, without loss of generality, are normalized such that $\sum w_i = 1$. Solving the above optimization problem for a certain number of different weight combinations yields a set of solutions.

Universiti Utara Malaysia

2.6.2 Related Past Work on Aggregating Multi-Objective Optimization

Aggregating functions is usually defined as the combining of objectives into a single function (Coello, 1999).

Weights were employed in the fitness function by (Syswerda & Palmucci, 1991) in order to increase or decrease values throughout the assessment of resource schedulers, decided upon by the absence or the presence of any penalties (violated restrictions).

Jakob, Gorges-Schleuter, and Blume (1992), in order to move the tool centre point of an industrial robot towards a fixed locality, utilised a weighted sum of several objectives in the task planning to ensure accuracy and speed, and to avoid certain barriers and strive to create a path that is short and smooth.

In 1993, Jones, Brown, Clark, Willett, and Glen (1993) employed weights for their genetic operators to indicate their efficiency during the application of a GA for the construction of hyper-structures from a set of chemical structures.

This method was incorporated into a GA by Wilson and Macleod (1993) and used to design non-multiplier infinite impulse response filters, whereby the two opposing objectives reduced the response error and the cost of implementing the filter.

This method was used by Liu, Begg, and Fishwick (1998) for the optimization of the outline and the positioning of the actuator of a 45-bar plane truss, whereby the objectives were to reduce the cost of the linear regulator quadratic control and to enhance the durability as well as the modal controllability of the control system in accordance with the total weight, asymptotical constancy, and eigenvalue limitations.

In 1994, a weighted sum method was employed by Yang and Gen to solve a bi-criteria linear transportation problem. This method was expanded recently by Gen, Ida, Li, and Kubota (1995) and Gen and Cheng (1997) in an attempt to rectify the uncertainty which can emerge during decision making, by the inclusion of greater than two objectives, as well a fuzzy logic. The fuzzy ranking technique was employed in combination with the weighted sum in order to identify the Pareto solution with coefficients of objectives referred to as fuzzy numbers, which denote the uncertainly surrounding their relative importance.

Universiti Utara Malaysia

Murata and Ishibuchi (1995) proposed a new approach for making a selection from a set of Pareto-optimal solutions. Their method was different from single-objective genetic algorithms in terms of the selection process and the elite preservation strategy. In their method of selection, the genetic algorithm picked individuals for a crossover operation according to a weighted sum of the multiple-objective functions. The distinctive feature of the selection process was that the weights attached to the multiple-objective functions varied according to the specifications of each selection.

According to Murata et al. (1996), MOGAs may be used for flow shop scheduling. Based on their earlier study, the distinctive characteristics of their algorithm were its selection process and elite preservation strategy. Individuals were selected by their MOGA for a crossover operation according to a weighted sum of multiple-objective functions with varying weights. Instead of a single elite solution, multiple elite solutions were employed by the elite preservation strategy in the algorithm. In other words, a specific number of individuals were picked from a provisional set of Pareto optimal solutions, and these were carried forward into the next generation as elite individuals.

Universiti Utara Malaysia

Clustering is essentially a complex problem because it involves the building of suitable objective functions and the optimization of the objective functions. Sheng, Swift, Zhang, and Liu (2005) proposed an objective function known as the weighted sum validity function (WSVF), whereby the suggested function is the weighted sum of several normalized cluster validity functions. In order to optimise the WSVF, a hybrid niching genetic algorithm (HNGA) was also introduced to automatically generate the correct number of clusters as well as the proper segregation of the data set. A niching technique was developed within the HNGA to maintain the diversity of both the population in terms of the number of clusters encoded in the individuals as well as the sub-population with an equal number of clusters throughout the search. Furthermore, the *k*-means algorithm was used to hybridize the HNGA. Both the HNGA and WSVF are effective on the whole in comparison to other related genetic clustering algorithms.

Ryu et al. (2009) introduced a new technique for estimating the Pareto front for the simulation of a multi-objective optimization problem (MOP) where the precise forms of the objective functions are unavailable. In the proposed method, each objective function was iteratively estimated by means of a meta-modelling system, and a weighted sum method was used to transform the MOP into a set of single-objective optimization problems. The weight on each single-objective function was adjusted based on access to newly-introduced points at the existing iteration and the non-dominated points. According to the results, evenly distributed points were effectively produced by the proposed algorithm for a variety of Pareto fronts.

Zou, Zhang, Yang and Gragg (2012) suggested a systematic method for obtaining a set of weights for forecasting upper body postures for 4 subjects with 18 targets for each subject. Eventually, an alternative method was developed by Zou et al. (2011a). Using the seated posture case, Zou, Zhang, Yang and Cloutier et al. (2011) extended this method to include standing reach tasks. In several researches, the global weights were computed by averaging the weights for all the subjects and tasks (Zou, Zhang, Yang and Boothby et al., 2011; Zou, Zhang, Yang and Cloutier et al., 2011; Zou et al., 2012).

Zou, Zhang, Yang, Cloutier, and Pena-Pitarch (2012) put forward a nonlinear inverse optimization method to ascertain the weights for the joint displacement function in standing reach tasks. The design variables were denoted by the weights of the cost function, which was the weighted sum of the differences between two sets of joint angles (predicted posture and actual standing reach posture). Three methods were used for the posture prediction, namely the empirical–statistical method, the direct inverse kinematics method, and the direct optimization-based method. In the first method, thousands of experimental data were gathered and processed by computer-aided software before being statistically analysed (Beck & Chaffin, 1992; Das & SENGUPTA, 1995; Faraway, Zhang, & Chaffin, 1999). This is a direct but rigid method. If the posture prediction settings were to change, a fresh experiment will have to be conducted. In the second method, a set of equations were employed to seek a solution (Griffin, 2001; Kim, Martin, & Gillespie, 2004; Tang, Cavazza, Mountain, & Earnshaw, 1999; Tolani, Goswami, & Badler, 2000; Wang, 1999; Wang & Verriest, 1998). Finally, in the third method, the minimum value of a cost function was generated by fulfilling all the limitation requirements. Certain performance measurements, such as discomfort , joint displacement Jung and Park (1994), Zou et al. (2011) and Zou et al. (2011a), can be used as cost functions for the formulation of multi-objective optimizations (Howard, Cloutier, & Yang, 2012; Yang, Marler, Kim, Arora, & Abdel-Malek, 2004). The main problem in multi-objective optimization is the determination of the relative significance of various human performance measures. Among the methods such as the weighted sum method, the min-max method and the global criterion method, which are employed to obtain

Pareto solutions to a MOO problem, the weighted sum method is the one that is more popular.

Weights are usually determined through trial and error (Athan & Papalambros, 1996; Ismail & Yusof, 2010; Messac & Mattson, 2002; Nedjah & de Macedo Mourelle, 2005; Yang et al., 2004) or by the self-adaptive weighted sum method (Khan, 2009; Kim et al., 2004; Kim & Weck, 2005; Ryu et al., 2009; Zhang, Han, Li, & Song, 2008), where the principal idea is to adjust the weights accordingly within a search area instead of adopting a priori weights or defining inequality constraints.

A third method is the consistency ratio method Saaty and Vargas (1991), where a hierarchy matrix is used to carry out comparisons of pairs in order to obtain the weights of all the factors before a consistency ratio, indicating the connection between the judgments and massive samples of purely random judgments, can be ascertained. The fourth method involves the use of a genetic algorithm to calculate the weights (Dong, Xu, Zou, & Chai, 2008; Rachmawati & Srinivasan, 2006). Zhang, Domaszewski and Fleury (2001) also introduced a weighting method consisting of the formulation of multi-bounds and convex programming for multi-criteria structural optimization (Zhang, Domaszewski, & Fleury, 2001).

The conventional weighted sum approach to multi-objective optimization looks for Pareto-optimal solutions one by one by methodically altering the weights of the objective functions. However, it has been proven in earlier studies that the identified method can produce a weak distribution of solutions on the Pareto line and fails to locate a Pareto-optimal solution at a non-convex area. An adaptive weighted sum (AWS) method

Universiti Utara Malaysia

proposed by Kim and Weck (2005) concentrates on unexplored areas by adapting the weights instead of utilizing a priori weight selection while stating additional inequality limitations. This method generates solutions that are well distributed, is capable of locating Pareto-optimal solutions in non-convex regions, while ignoring non-Pareto optimal solutions. The method was further expanded through the development of a bi-objective AWS method for problems that have more than two objective functions (Kim & Weck, 2006).

Rachmawati and Srinivasan (2006) came up with a method that calculates the conversion of original objectives in accordance with the weighted sum functions. The converted functions are able to detect niches that will coincide with knee regions within the objective space. The niche strength as well as the parameters of the pool size control the density and extent of coverage within the knee regions. Although the algorithm is based on weighted sums, it is able to locate a possible solution within the non-convex area of the Pareto front. Favourable results have been obtained through the use of an algorithm which tests identified problems within several knee areas and skew on the Pareto-optimal front.

Messac and Mattson (2002) developed a physical programming-based method to generate the Pareto front. In this method, the behaviours of the objective functions were examined in terms of their respective abilities to navigate in the design space, or equivalently, the ability to generate well-distributed sets of Pareto points. In particular, the behaviours of the weighted sum, as well as compromise programming, and the physical programming methods were examined.

2.7 Past work in Nonlinear Knock Factor Optimization and Evaluation Function

Attar and Karim (1998) combined a deterministic gradient-based model with a simple genetic algorithm to predict the analytical parameters (variables) necessary for an engine to work within the optimum performance zone. This method was replaced by a restricted optimization problem and by other non-restricted problems to facilitate the formulation of the problem. Researchers used objective functions, such as the exterior penalty function with constraints, which were multiplied by the control coefficient *r*, and in turn were used to determine the magnitude of the exterior penalty function. The minimization must start with a relatively small value of *r*, which is gradually increased (Haftka & Gürdal, 1992).

Khalil, Camal, and Laurent (2009) extended the previous model Attar and Karim (1998) to evaluate thermodynamic model assumptions that deal with the factor and methane number in gas, which affect the knock.

Douaud and Eyzat (1978) examined the behaviour of knocking by using the factors that affect the knock, namely temperature and pressure, which were represented by a nonlinear mathematical model. They used this model to calculate the values of the pressure and temperature coefficients, as well as to reduce the total sum of squares of deviation present between integration and theoretical values for *N* tests.

Mockus (2002) presented the basic ideas of an updated method known as the Bayesian heuristic approach (BHA), and explained the application of the Bayesian Approach (BA) to heuristic optimizations.

The BHA aims to fix prior distributions, P to a set of auxiliary functions $f_k(x)$ that can describe the best values gained through the use of K times some heuristic $h(x)$. The heuristic $h(x)$ is supposed to optimize an original function, $v(y)$ of variables $y \in \underline{R^n}$ (Mockus et al., 1996).

Al-Duwaish (1997) came up with a novel method for controlling nonlinear dynamical systems through the use of genetic algorithms. The proposed non-linear controller was designed to locate the optimal control input sequence, which reduces the error that may be present between the output of a nonlinear system and that of a reference model. A genetic algorithm was employed for the optimization process due to its ability to substantially shrink the input search space. Although the single input/single output concept was used, this method can just as easily be used for multi input/multi output (MIMO) systems (Al-Duwaish, 1997).

Universiti Utara Malaysia

di Gaeta, Giglio, Police, Reale, and Rispoli (2010) also evaluated knocking based on a pressure signal inside the cylinder by creating a model based on a partial differential equation. This equation differs from the classical wave formulation adopted by (Draper, 1934). Draper's approach provided a simple analytical method to calculate the resonance frequencies and the vibration modes inside an engine cylinder.

Recently, Spelina, Jones, and Frey (2014) presented a modern method in signal analysis and simulation accuracy and control. The statistical properties of knock density and knock events show that the knock density behaves as an independent random process, and the knock events follow a binomial distribution. These properties have a major effect on the simulation and control of the knocking. They dealt with the statistical properties of knocking to re-tune one factor that affects knocking, which is the ignition timing.

Bozza et al. (2014) studied the behaviour of knocking in an SI turbocharged engine at drive-in conditions of full conversion. They also proposed a method that shows interesting advantages in terms of higher accuracy and sensitivity compared with the classical maximum amplitude of pressure oscillation approach (Brecq, Bellettre, & Tazerout, 2003). The authors measured a series of 200 consecutive pressure cycles for each speed and selected the standard engine calibration spark advance. They used two methodologies to analyse knocking under experimental and numerical points. The first was an auto-regressive model for the cylinder pressure signal. After the initial configuration, the knock index was calculated and the noise variance (NV) for a series of pressure cycles was obtained. They also used a technique to select an appropriate model for control. This technique, called the Akaike Information Criterion (AIC), is characterized by simplicity in model, and requires only a few adjustable parameters to ensure accuracy.

Numerous factors such as volumetric efficiency, residual gas, air–fuel ratio, charge motion, combustion chamber temperature and so on contribute to knocking. In 2003, the Japanese Honda company Shih, Itano, Xin, Kawamoto, and Maeda (2003) conducted a study on the working combustion temperature inside the cylinder. The results showed that the uniformity of the individual cylinders needed to be improved by reducing variations in the combustion chamber temperatures among all the cylinders.

Jagtap Harishchandra, Koli Ravindra and Baste Sachin (2010) attempted to reduce the tendency of knocking by optimizing the charge temperature at the end of the compression stroke, and they effectively suppressed the knock by optimizing the flow of coolant through the cylinder head and crankcase.

Through optical diagnostics, Merola et al. (2011) analysed the knock pressure signal to optimize the spark ignition.

To avoid knocking, Elmqvist, Lindström, Ångström, Grandin, and Kalghatgi (2003) developed a simulation model for the control of phases in turbocharged engines. They monitored and optimized this model through a series of tests for different speeds and lambdas.

Dao (2010) conducted a study on the modelling of the chemical dynamic engineering process to analyse, optimize, and control the behaviour of dynamic systems. A nonlinear mathematical model was developed to define the dynamic behaviour of a continuous stirred tank reactor (CSTR). The researcher used evolutionary algorithms in the field of artificial intelligence, and optimized and controlled the chemical reactor processes by using a genetic algorithm (GA) on different sets of factors or cost functions. The area resulting from the difference between the needed and actual temperature profiles of the reaction mixture within a selected time period, known as the CSTR cycle duration, was reduced by such an optimization.

2.8 Discussion and Summary

Literature related to the area of research has been outlined within this chapter. The discussion included the basic concepts, techniques, previous works and problems of optimization. Previous works in relation to optimization, multi-objective optimization, and aggregate multi-objective optimization have been discussed. A further review of the literature on the nonlinear knock factor optimization and evaluation function was also conducted.

Based on the summary of the algorithms, some overlaps may exist between the algorithms mentioned above. Furthermore, the combination of these research strategies is often desirable and possible, leading to the issue of non-trivial design search algorithms. Generally, robust optimization has to identify a trade-off between the quality of solutions and their robustness in terms of decision variable disturbance. This problem may be formulated as a multi-objective optimization problem. Unlike optimization under uncertainty, the objective function in robust optimization is considered as deterministic. If multi-objective optimization problems have concave Pareto fronts, weighted sum (WS) approaches tend to fail to find entire Pareto fronts (i.e., all the Pareto optimal solutions). Our approach, however, can handle multi-objective optimization problems with concave Pareto fronts.

A nonlinear weight selection methods reviewed in this study has been shown to provide a means of controlling the distribution of points on the convex Pareto front. One major drawback of the WS method is that it does not provide means of controlling the distribution of points on the Pareto front. This is due its inability to take into consideration the curvature of the Pareto surface to determine its own slope change and also to control its own slope sensitivity. This is because the weight space constraint for the standard WS is defined on a simplex which does not have curvature. The proposed method maps the nonlinear weight space into another form space constraint which allows its curvature to be controlled through free parameters.

From a purely mathematical point of view, even the 'simpler' global optimization (GO) model instances, for example, concave minimization, or indefinite quadratic programming belong to the hardest class of mathematical programming problems. The computational difficulty of any general class of such models can be expected to increase exponentially, as a function of the problem dimensionality n.

Universiti Utara Malaysia

CHAPTER THREE
RESEARCH METHODOLOGY

3.1 Introduction

This chapter presents a new approach for developing aggregate nonlinear evaluation function techniques used to optimize Nonlinear Multi-objective Optimization Problems(NMOOPs) by constructing a new evaluation function and using it as a fitness function for optimization in genetic algorithm (GA). This work consists of three main phases: data gathering, objectives modelling, and optimization. Data gathering shows the data type and how collect this data. Objectives modelling consist of two stages, system identification (SI) and the aggregation of the objective functions. System identification (SI) which included observed data, estimation, complexity, validation and evaluation of the model. Aggregation of the objective functions includes two steps, the individual objective functions are aggregated, and the "partial derivative" (PD) is applied to the aggregated individual objective function "Evaluation Function". While optimization includes apply continuous GA in order to obtain outcomes. Figure 3.1 and the sections below illustrate these phases.

Universiti Utara Malaysia

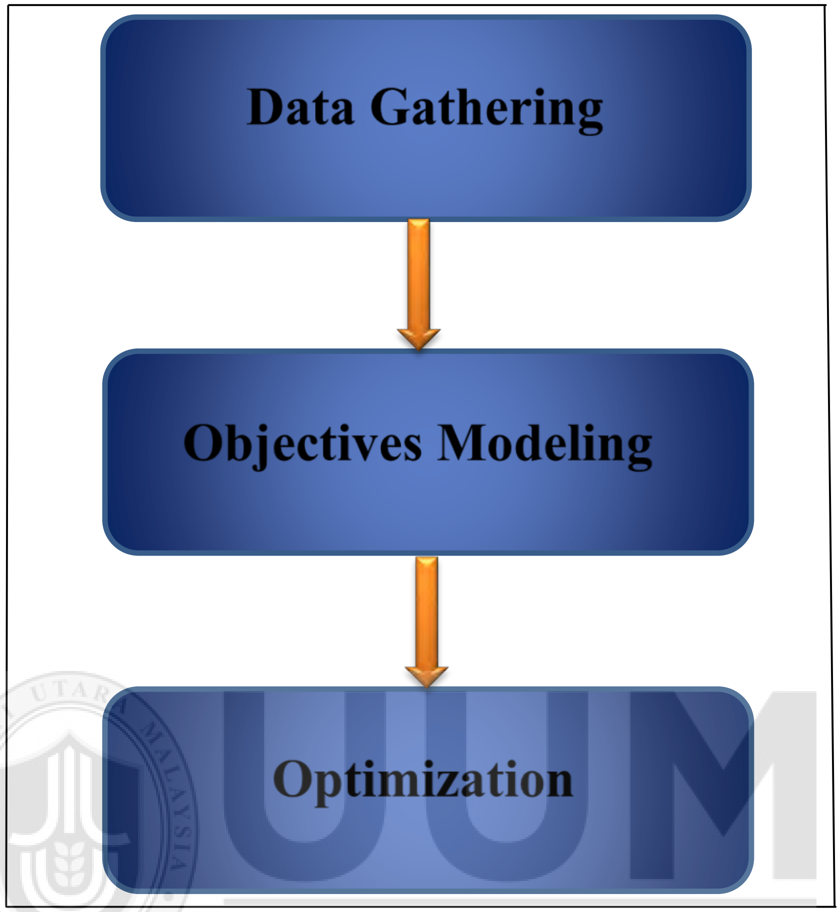


Figure 3.1. Framework for the optimization of NMOOPs

Universiti Utara Malaysia

3.2 Phase One: Data Gathering

Concept of data fusion technology for multi-sensors, used in this study, which refers to a combination of data and information from multiple sensor, to obtain a more accurate assessment of a specific problem and identify the problem associated with these threats Ma (2001) (a combination of data and information from multiple sensors), consequently achieving improved accuracy and obtaining the best inference on the problem under discussion. The concept of multi-sensors data fusion is a very modern one. The evolution of animals and humans has led to the ability to utilize multiple senses in order to increase their chance of survival. A good example of integration of data systems can be seen within the human or animal brain. The brain is capable of integrating sensory information, such as sound, smell, visual, taste, as well as concrete data in order to achieve a complex understanding of the environment around them, which ultimately increases the chance of surviving within that environment.

In this work focused on parametric multi-sensors data. Parametric algorithms are based on the assumptions of a parametric model, which consists of fitting the data model and estimating model parameters. In contrast, nonparametric algorithms are not based on any model parameters. Thus, the nonparametric algorithm is applied when problem parameterization is unknown or unavailable (Ma, 2001).

One of the difficulties in this work is to determining which data type should be used in terms of "gathering" to access to the best modelling that represents the problem. According to the nature of the problem at hand, therefore should be using data overlapping gathered by manner namely "multi-sensors data fusion". Generally, complex systems such as internal combustion engines have multiple sensors embedded at various levels within their structure. Sensors are data gathering mechanisms that measure a systemic quantity (such as functionality or failure) that provides the researcher or engineer with a multitude of reliability information. When data sets are drawn simultaneously from multiple sensors in a system, they are said to be overlapping or a "fusion" data set as a result of the mutual influence between them (Jackson, 2011).

By identifying less value and the highest value for each of the objective functions, has been determined the appropriate work space for decision variable (knock).

$$\boldsymbol{min \leq\ Tps \leq max}\ ,$$

$$\boldsymbol{min \leq\ Rpm \leq max \quad and}$$

$$\boldsymbol{min \leq\ Temp \leq max}$$

A number of multi-objectives geometric problems require such methodology for data collection to obtain more reliable data for process modelling and testing when solving problems. Data were also gathered from sensors that allow analysis of overlapping data sets. The inherent inter-dependence of these data sets was exploited to yield significant additional information. In our problem, the data collection methodology was also exploited to obtain the greatest possible reliability in representing the proposed model.

Universiti Utara Malaysia

3.3 Phase Two: Objectives Modeling

The stages of modelling will be described in this phase. Two methodologies were used to construct the model. The first is system identification (SI). In this stage, the construction of the model for evaluation utilized the same research methodology as Dym (2004) and Ljung (2010), which included observed data, estimation, complexity, validation, and evaluation of the model. The second methodology is the aggregation of the functions.

According to Carrejo and Marshall (2007), in the beginning, Newton's mathematical methodology, "Newtonian style," was used, particularly in the Principia. Three main elements combine within the mathematical modelling process, which requires prior experience perceptions or the modelling process which may not resemble common concepts presented in mathematics or physics, See Figure 3.2.

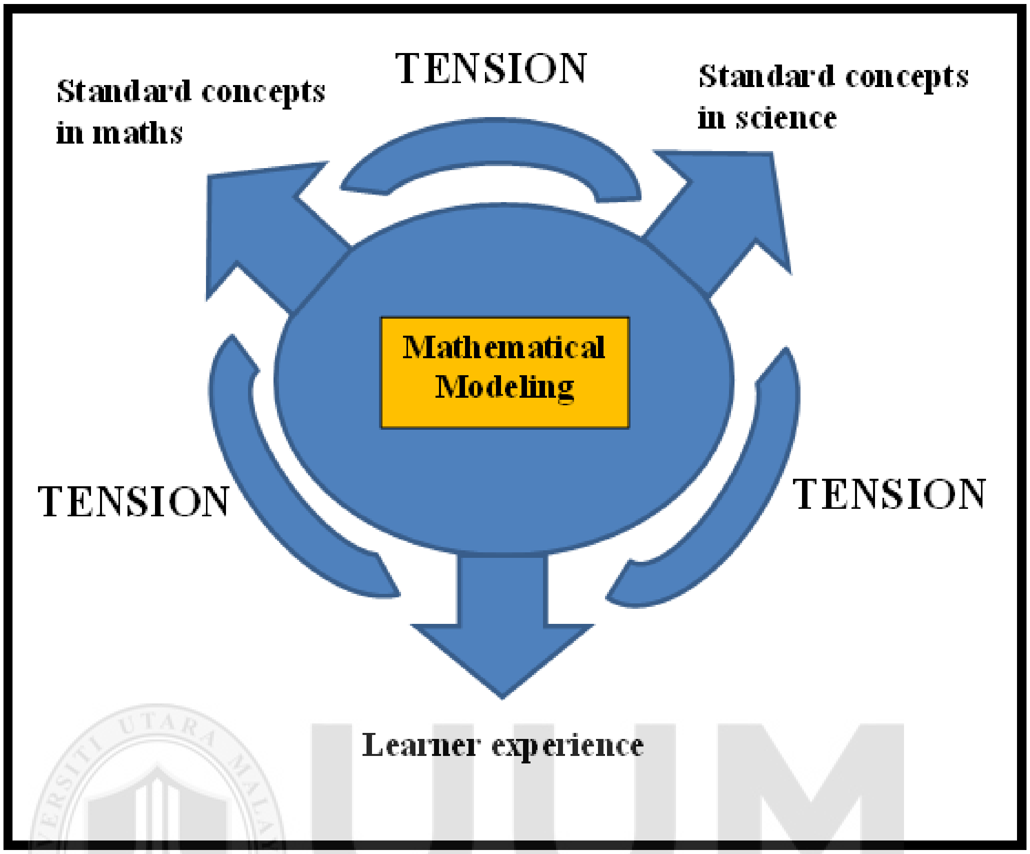


Figure 3.2. Tensions during the mathematical modeling process (Carrejo & Marshall, 2007)

The learners' real world experience may present cognitive conflicts and knowledge tension during contextual inquiry and during the study of standard mathematical and physical concepts. The level of complexity required in the construction of a mathematical model capable of describing and predicting the motion of an object was revealed in the results of the investigation. Perceptions which are described within the previous paragraph are also necessary. For example, Figure 3.3 shows the outcomes of the element interactions in the modelling problem.

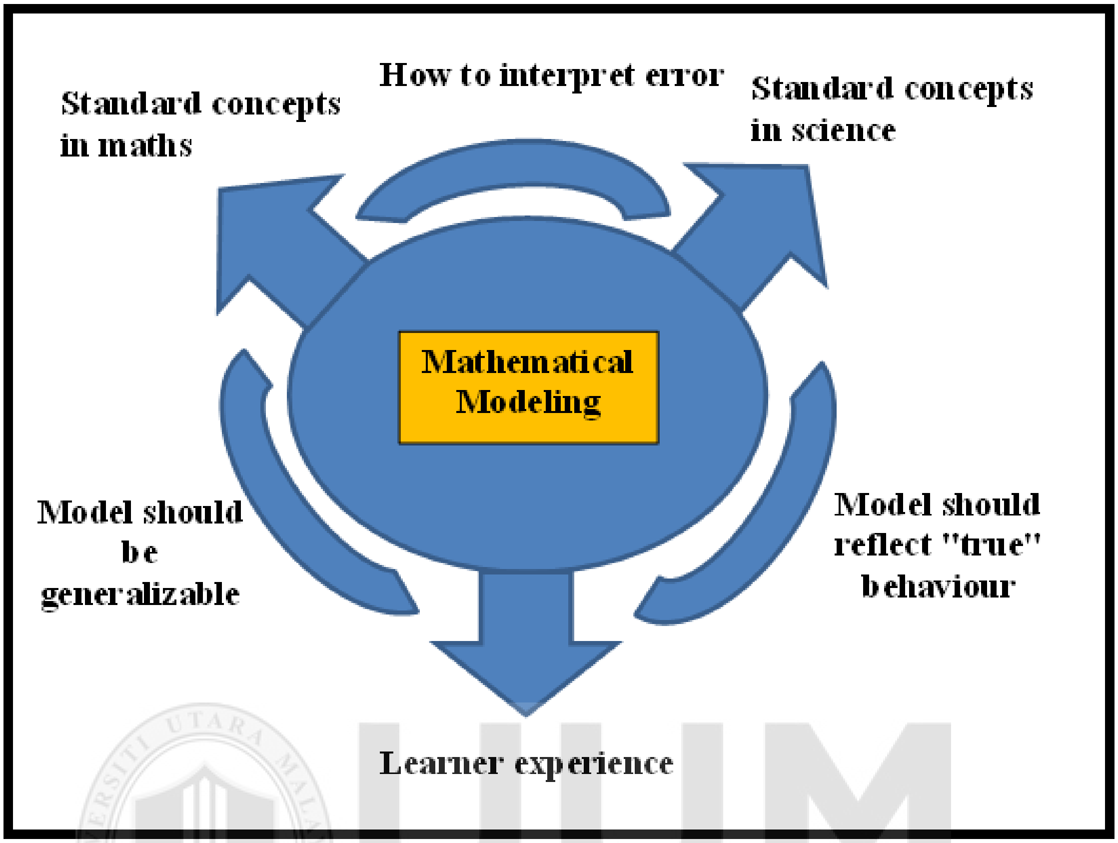


Figure 3.3. Summary of tensions from hypothetical experiment (Carrejo & Marshall, 2007)

Universiti Utara Malaysia

Specifically, Figure 3.4 shows the tensions related to the data and the scale function.

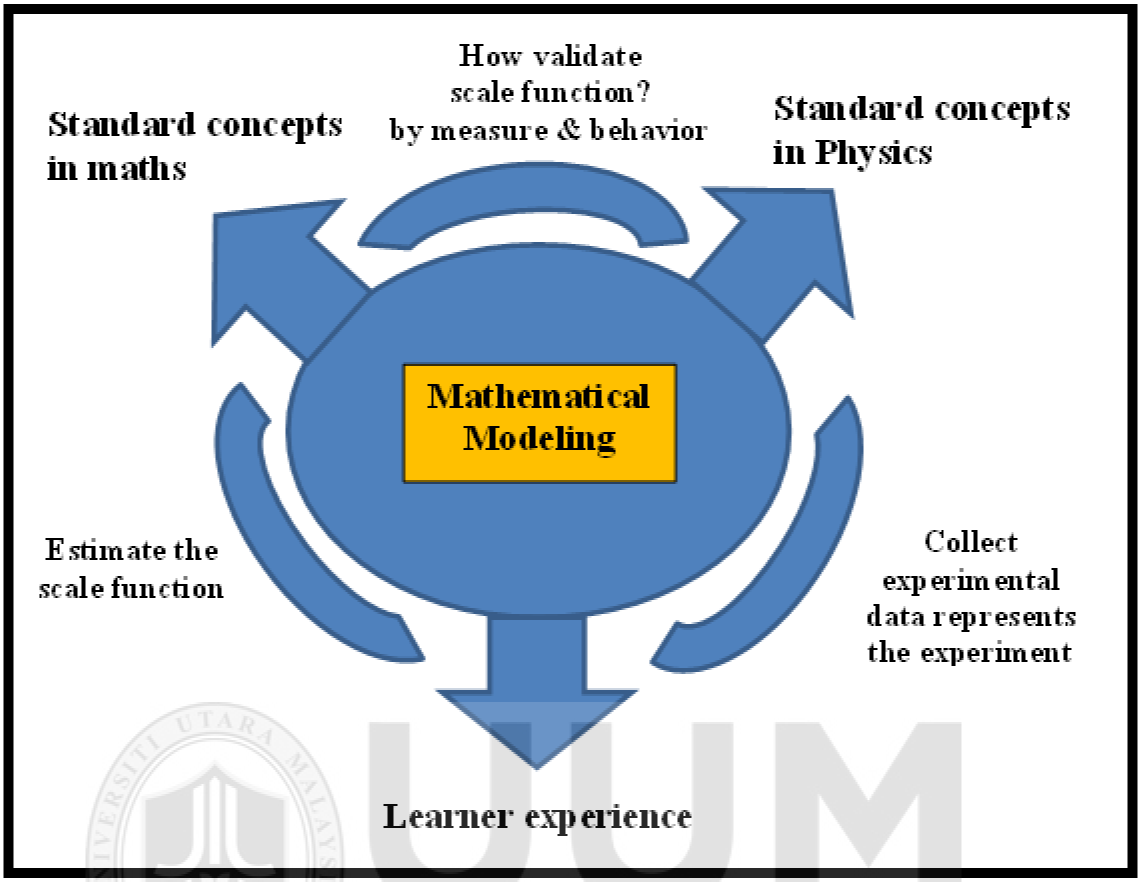


*Figure 3.4.*Summary of tensions related to data and scale function (Carrejo & Marshall, 2007)

3.3.1 System Identification

The first methodology applied System identification, SI often refers to the construction of models within the mathematical context using observed input-output data. SI is seen as the connection between real-world applications and the mathematical world containing control theory and model abstractions (Ljung, 2010). See Figure 3.5.

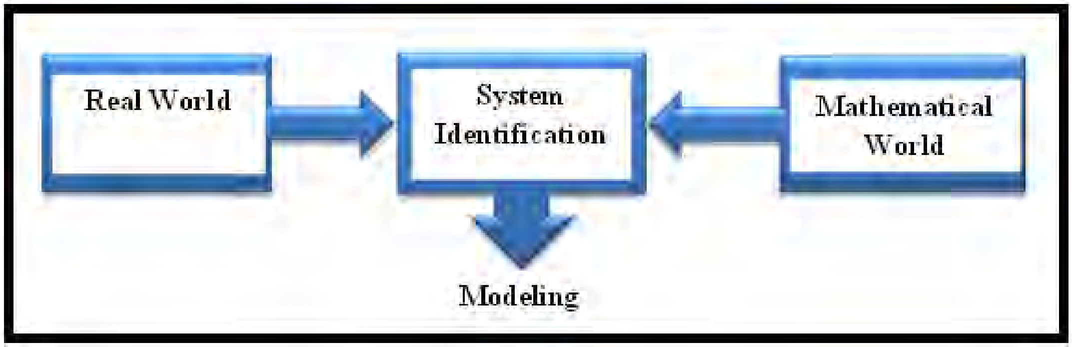


Figure 3.5. Concept of SI (Ljung, 2010)

In the past decades, a trend toward nonlinear modelling has been observed in different application areas. Technological innovations have resulted in fewer restrictions on computational memory and access to data, making non-linear modelling a more appropriate option. Various method of SI to build models employed for the nonlinear factors studied. The primary goal of the system is to identify mathematical models with available input–output data. This goal is often achieved by minimizing cost function, which is an integral part of the statistical framework, see Figure 3.6, (Dahleh, 2011; Paduart, 2008).

Universiti Utara Malaysia

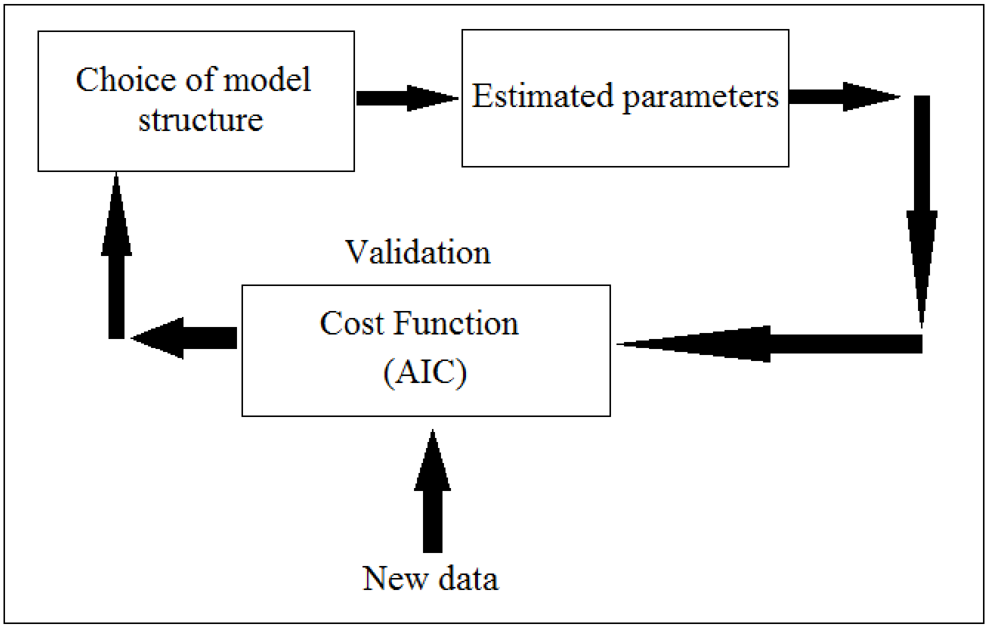


Figure 3.6. Basic idea of SI: cost function relates data and model (Dahleh, 2011)

Nonlinear SI is able to divide into two cases: off-line and on-line (adaptive). In this study, focusing on the off-line case will be useful for gaining insights into the identification problem.

Universiti Utara Malaysia

An important step in the identification procedure is the estimation of the parameters in the model (Aarts, 2011). Initially, a model can be defined as the relationship that exists between the observed quantities within the system, such as (TEMPERATURE-KNOCK). Roughly speaking, a model can allow a prediction of properties or behaviours for an object. Usually, this relationship can be represented as a mathematical expression, table, or graph.

With input (x) and output (y) the main objective is to identify the nonlinear system *m*. Identification usually refers to selecting the optimal model *m* from within the class or family of models *M*. “Best” can be measured using a selected criterion or cost function.

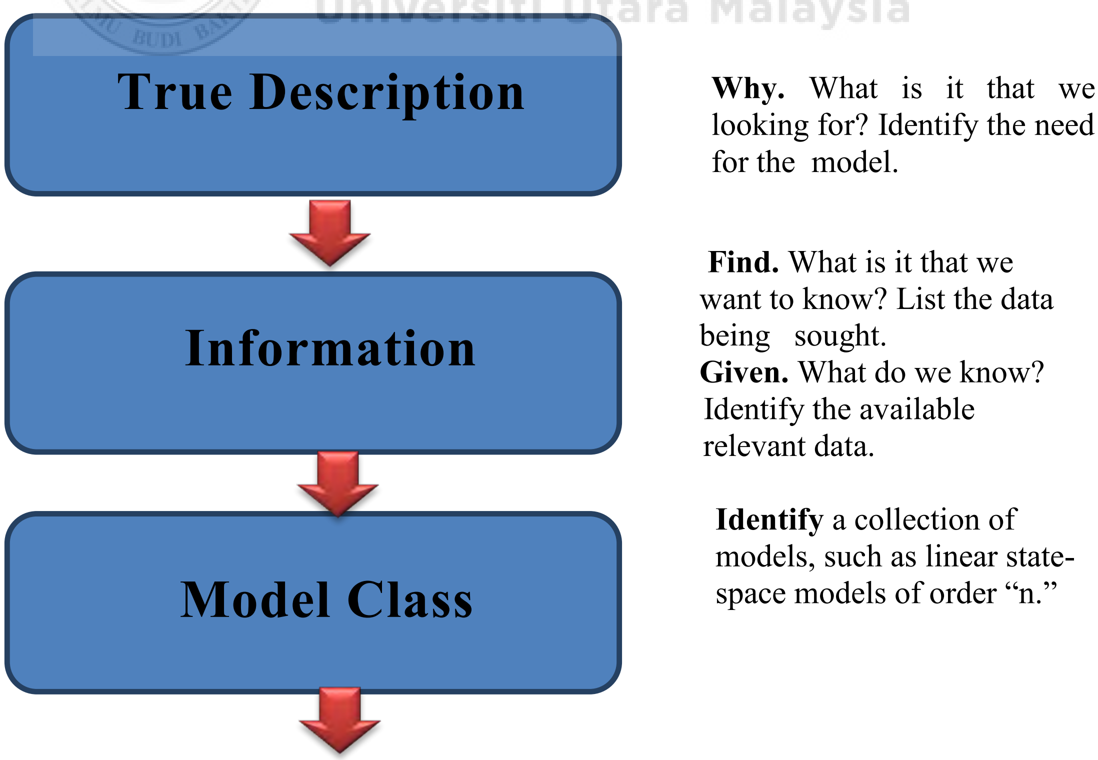
Several typical cost functions can be used. The first expression is Akaike's Final Prediction Error (AFPE) whereas the second is Akaike's Information Criterion (AIC) and the third is the generalized cross-validation criterion (GCVC) (Ljung, 2010). The total sum of squared errors of outputs can be predicted using the model and measured outputs (Nowak, 2002). Also note the criterion by (Cost) *C*. The nonlinear SI problem can be mathematically presented as the following equation:

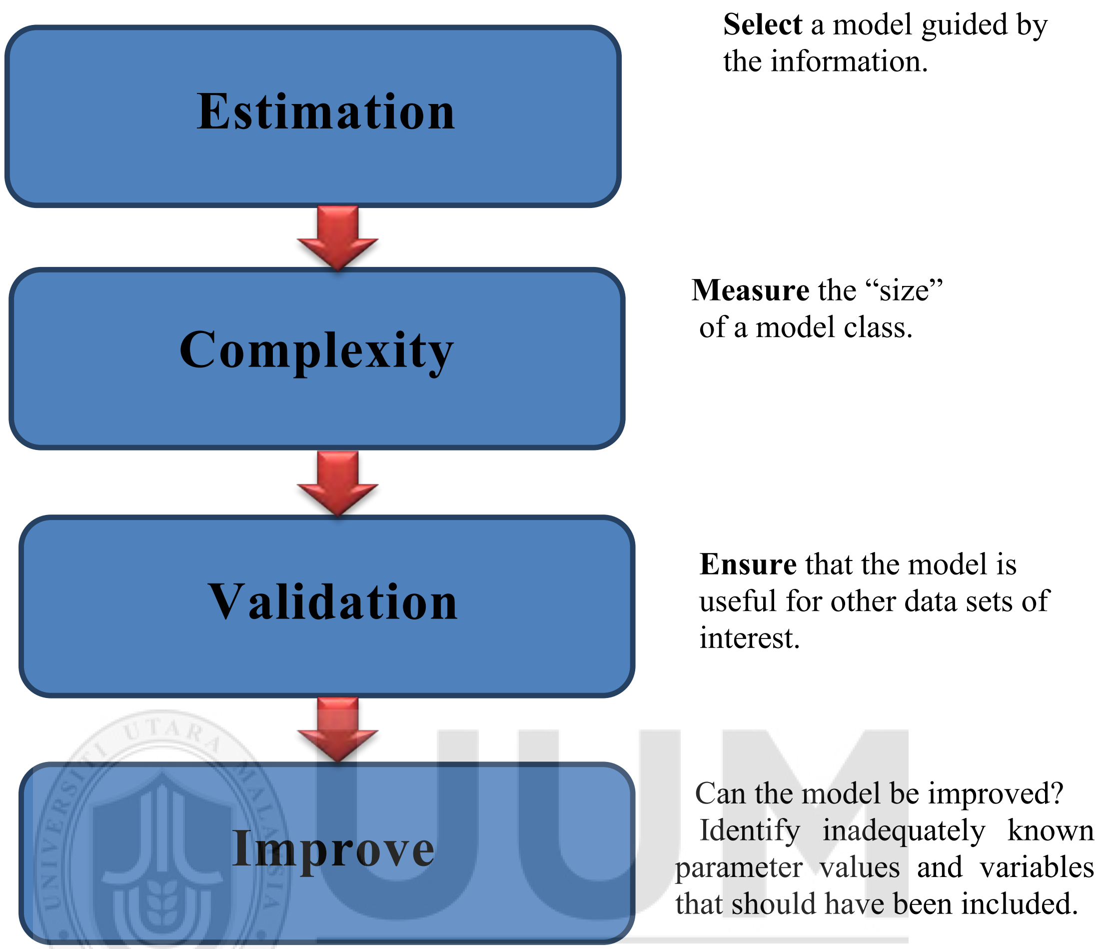
$$m' = \min_{m \in M} C(x, y, m) \ ,$$

where *m* is chosen from the class *M* which minimizes the cost *C* (*x, y, m*). The most crucial element of nonlinear SI is the accurate specification of the family *M* (Nowak, 2002).

Universiti Utara Malaysia

In general, the SI steps can be summarized as in the Figure 3.7 below:





Universiti Utara Malaysia

Figure 3.7. Summary of system identification steps (Ljung, 2010)

True description: In most cases, the true description of the modelled object does not realistically achieve "true" description. Hence, assuming such a description as an abstraction is often more convenient because while it has the same characteristics as the model, the description is typically much more complex.

Information: This step includes information obtained from the observed data as well as prior information regarding the object to be modelled, for example the model class.

Model class: This step will identify the model class, which refers to a collection of models. A model class may refer to the set that can be parameterized by the finite-dimensional parameter, such as “all linear state-space models of order n”.

Estimation (learning): This step refers to a process which can select the model guided by information. Referring to this model as process learning has become also increasingly common among statisticians. The elementary ***curve-fitting technique*** could be useful in dealing with the estimation problem.

Model fit: The extent of which a certain model m can explain or fit to a certain data set Z in a (scalar) measurement. This is presented as F(m,Z).

Complexity: The measurement of size or flexibility. Complexity measurements are denoted with the symbol AIC. It may be used as the vector which parameterizes a set in an efficient manner.

Validation (generalization): This may be used to gauge the generalizability of the algorithm, or to conduct an estimation of performance potential of a learned model using current data from one algorithm. The model must be useful for the estimation data as well as other data relevant data sets. These data sets are known as ***validation data*** (testing data). An alternate name for this process is ***generalization*.** Finding a model that describes estimation data well is not very difficult. With a flexible model structure, finding something well-adjusted to the data is always possible. The estimated model can be presented with a new set of data and this is where the real test becomes apparent. The average fit to validation will become inferior to the fit to estimation data. Multiple analytic results confirm this deterioration of fit.

Usually, the model is constructed via the consideration of two concepts. Firstly, the proposed model should display strong agreement with the estimation data and secondly, it should not be overly complex (Ljung, 2010). The concepts are slightly contradictory, and therefore a compromise should be sought, which will be further discussed in the following section. The (m) model also becomes a random variable as the information is typically described by random variables (Ljung, 2010).

3.3.2 Aggregation Evaluation Function Methodology

The second methodology applied consist of two steps will be performed in this stage. First, the individual objective functions are aggregated. Second, the "partial derivative" (PD) is applied to the aggregated individual objective function "Evaluation Function", as shown in the Figure 3.8 below:

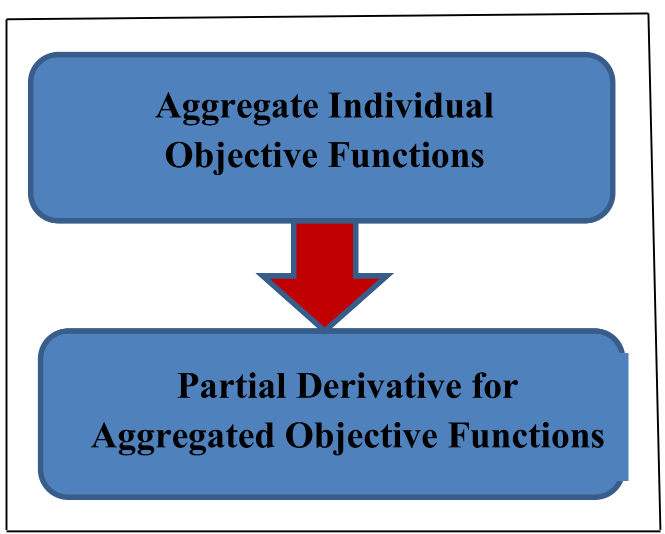


Figure 3.8. Aggregation evaluation function methodology

3.3.3 Differential calculus and derivatives

This section discusses the partial derivatives used to find a partial effect for each objective in the model of objective function, as well as the rate of change of one quantity in respect to another (Horan & Lavelle, 2005; Thomas, 1996).

3.3.4 Partial Derivatives

Assume a real-valued function $z = f(x, y)$ of two real variables. The derivative of f with respect to x holding y constant is called the partial derivative of f with respect to x and is denoted by $\partial f/\partial x$. Similarly, the derivative of f with respect to y holding x constant is called the partial derivative of f with respect to y and is denoted by $\partial f/\partial y$ (Horan & Lavelle, 2005).

In other words, it is supposed that f is a function of two or more independent variables. At every point in the domain, the function may possess differing instantaneous rates of change, in different traced directions. These **directional derivatives** are able to be computed utilising the instantaneous rates of change of f along the directions of the coordinate axes (of independent variables). The rates of change along these "principal directions" are known as the **partial derivatives** (PD) of f.

The usual rules of differentiation can be applied in order to find the partial derivative of f in respect to x for a function of two independent variables $f(x, y)$, see Table 3.1.

There is a single exception in that when the second variable y becomes apparent, it is treated as a constant. Also, the partial derivative of f with respect to y is able to be found by treating x as a constant when it becomes apparent.

Table 3.1

Rules of differentiation

General Formulas

$\frac{d}{dx}c = 0$	Constant Rule
$\frac{d}{dx}cx = c$	Factor Rule
$\frac{d}{dx}(cu) = c\frac{du}{dx}$	Factor Rule
$\frac{d}{dx}(u \pm v) = \frac{du}{dx} \pm \frac{dv}{dx}$	Sum Rule
$\frac{d}{dx}(uv) = \frac{du}{dx}v + u\frac{dv}{dx}$	Product Rule

$\frac{d}{dx}(uvw) = \frac{du}{dx}vw + u\frac{dv}{dx}w + uv\frac{dw}{dx}$	Product Rule
$\frac{d}{dx}\left(u/v\right) = \frac{v\frac{du}{dx} - u\frac{dv}{dx}}{v^2}$	Quotient Rule

$\frac{du}{dx} = \frac{du}{dv}\frac{dv}{dx}$	Chain Rule
$\frac{d}{dx}x^n = nx^{n-1}$	Power Rule
$\frac{d}{dx}u^n = nu^{n-1}\frac{du}{dx}$	Power Rule
$\frac{du}{dx} = \frac{1}{dx/du}$	

Trigonometric Functions

$\frac{d}{dx}\sin x = \cos x$	$\frac{d}{dx}\cos x = -\sin x$
$\frac{d}{dx}\tan x = \sec^2 x$	$\frac{d}{dx}\cot x = -\csc^2 x$
$\frac{d}{dx}\csc x = -\csc x \cot x$	$\frac{d}{dx}\sec x = \sec x \tan x$

UUM
Universiti Utara Malaysia

This method can also be used for a function containing more than two independent variables. As an example, a partial derivative of the function with respect to the variable x is able to be obtained via differentiation with respect to x utilising the usual rules for differentiation. The exception is that other independent variables are treated as constants when they occur in the expression of f.

With a function containing a single variable $y = f(x)$, a change in the independent variable x will lead towards a corresponding change in dependent variable y. The rate of change of y with respect to x is depicted using a derivative (df / dx). For function with more than one variable, a similar situation will occur. In the relation $z = f(x, y)$, x and y are the independent variables, and z is the dependent variable.

This method is used as x and y will vary, as will the z value. The x and y may simultaneously change, which triggers a change in z. Instead of considering this general situation, one of the independent variables will be fixed, which is equivalent to moving along a curve obtained by intersection the surface by one of the coordinate planes.

Consider $f(x, y) = x^3 + 2x^2y + y^2 + 2x + 1$. If we keep y constant and vary x, then the rate of change of the function f needs to be found. If we hold y at the value 3, then

$$f(x, 3) = x^3 + 6x^2 + 9 + 2x + 1.$$

In effect, we now have a function of only x. If we differentiate it with respect to x, we obtain the expression:

$$3x^2 + 12x + 0 + 2 + 0 \equiv 3x^2 + 12x + 2.$$

We say that f has been differentiated partially with respect to x. We denote the partial derivative of f with respect to x by $\partial f / \partial x$ (to be read as 'partial dee f by dee x'). In this particular example, when $y = 3$,

$$\partial f/\partial x = 3x^2 + 12x + 2.$$

In the same way, if y is held at the value 4, then $f(x, 4) = x^3 + 8x^2 + 16 + 2x + 1$, and thus for this value of y,

$$\partial f/\partial x = 3x^2 + 16x + 2.$$

If $y = c$, a general constant, then

$$f(x, c) = x^3 + 2x^2c + c^2 + 2x + 1,$$

and partial differentiation yields the expression

$$\partial f/\partial x = 3x^2 + 4cx + 2.$$

Now, if we return to the original formulation

$$f(x, y) = x^3 + 2x^2y + y^2 + 2x + 1,$$

and treat y as a constant, then the process of partial differentiation with respect to x gives

$$\partial f/\partial x = 3x^2 + 4xy + 0 + 2 = 3x^2 + 4xy + 2.$$

Universiti Utara Malaysia

Finally, can describe the final form of the evaluation function as an aggregate (sum) of partial derivative functions of each function representing one objective in the optimization problem, see Figure 3.9.

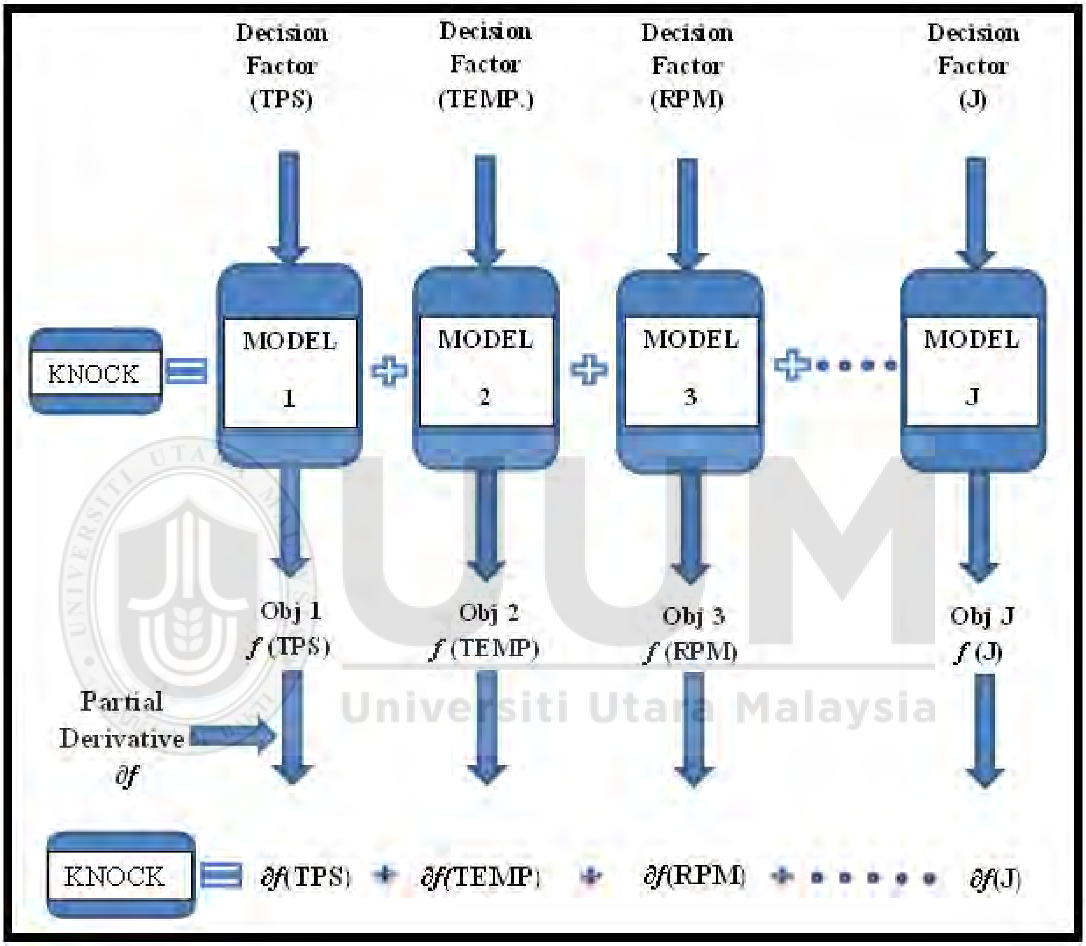


Figure 3.9. Basic formulation of multi-objective evaluation function (Tarun, 2008)

Differential calculus is about precisely describing the ways in which related quantities change.

In part above (Figure 3.9) of the methodology, the overall evaluation function was calculated by using the mathematical concept of “partial derivative” combined with aggregate objective functions.

This concept can be explained by the situation where the function of two or more independent input variables will change in at least one of the input variables, and therefore the change in the function must be calculated. Partial differentiation refers to the holding of all, except one, variable constant and then finding the rate of change of the function in regards to the remaining variable (Thomas, 1996).

3.4 Phase Three: Optimization Methodology

This part describes the optimization process methodology by using a flowchart illustrated in Figure 3.14.

Many MOOPs have appeared in our lives in recent decades. GA is the one of most timely approaches for multi-objective optimization subjects. Before discussing the optimization process methodology, should explain why used a GA in MOOPs.

3.4.1 GA in MOOPs

In the past, the complexity of the factors is specified, so using the model of a single objective and multi-objectives of such cases was possible. In the last two decades, however, two advances made modelling the best option in such a case. The first advancement is a significant increase in computing power and consequently in the speed of the solutions made available to the modeller. The second is the development of more than one algorithm to address such models. The principle advanced here has been in the area of meta-heuristics. A heuristics is defined by Luke (2015) and Reeves (1995) as a technique that seeks or finds a good solution to a difficult model. The most popular meta-heuristics include GAs (Golberg, 1989; Luke, 2015), which simulate species breed in the field of genetics; annealing, which mimics the cooling of materials in the field of physics; and the Tabu search, which emulates the concept of social "Taboo."

The multi-objective problem is one of the most promising directions in the context of GA application. The product has the potential to complement many daily life applications as well as address several engineering design issues (Konak et al., 2006). This potential is only beginning to be realised as it is capable of simultaneously supporting several, contrasting solutions of a problem, which allows the designer a greater range of choice and flexibility.

Universiti Utara Malaysia

The GA is not likely to offer competitive results for problems that are very well understood, almost linear, and has a reliable solution. If the problem can be solved analytically, then an acceptable level of hypotheses may be the best approach. However, if no solution to such problems exists, then GA can prove to be important and beneficial.

Several possible solutions can be derived for a problem using GA, with a final selection being taken by the user. Sometimes, a situation arises where there is not one single solution, such as a family of Pareto-optimal solutions. In a case where there are multi-objective optimization and scheduling problems, GA is able to simultaneously identify alternative solutions.

The highly accurate description is able to guide the decision maker to refine the requirements. The decision maker is able to gradually reduce the size of the solution while identifying the trade-offs between objectives in a case where objectives are able to be supplied interactively at each GA generation. The objectives variability doesn't impose constraints on the search space while it acts as a changing environment to the GA. The size of the solution set changes in a way that is known by the decision maker and therefore the appropriate sharing coefficients are still able to be calculated (Fonseca & Fleming, 1993). In general, a multi-objective genetic optimizer can be described as in Figure 3.10.

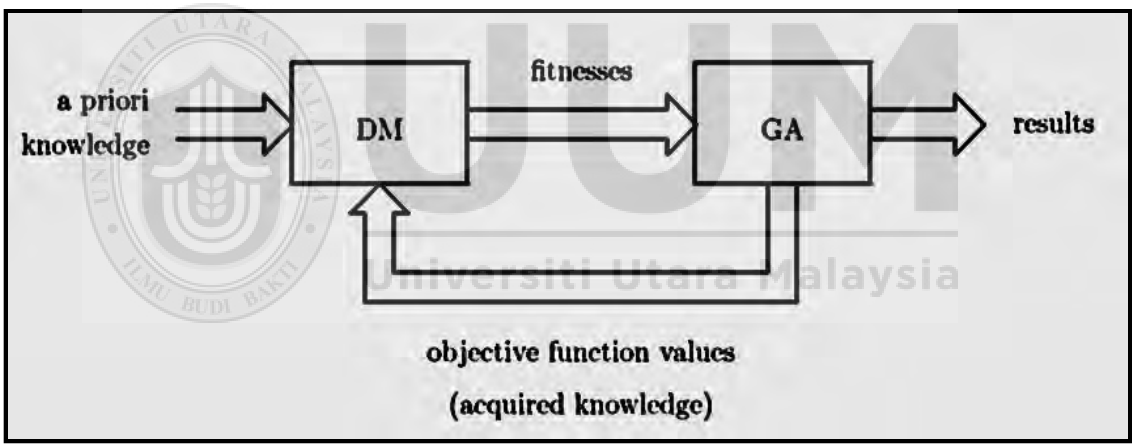


Figure 3.10. A general multi-objective genetic optimizer (Fonseca & Fleming, 1993)

As shown in the flowchart shown below, Figure 3.11, in optimization methodology, a new aggregate evaluation function model (AEFM) can be used as a fitness assignment procedure. As one of the most important operators, this model influences the effectiveness of multi-objective GA.

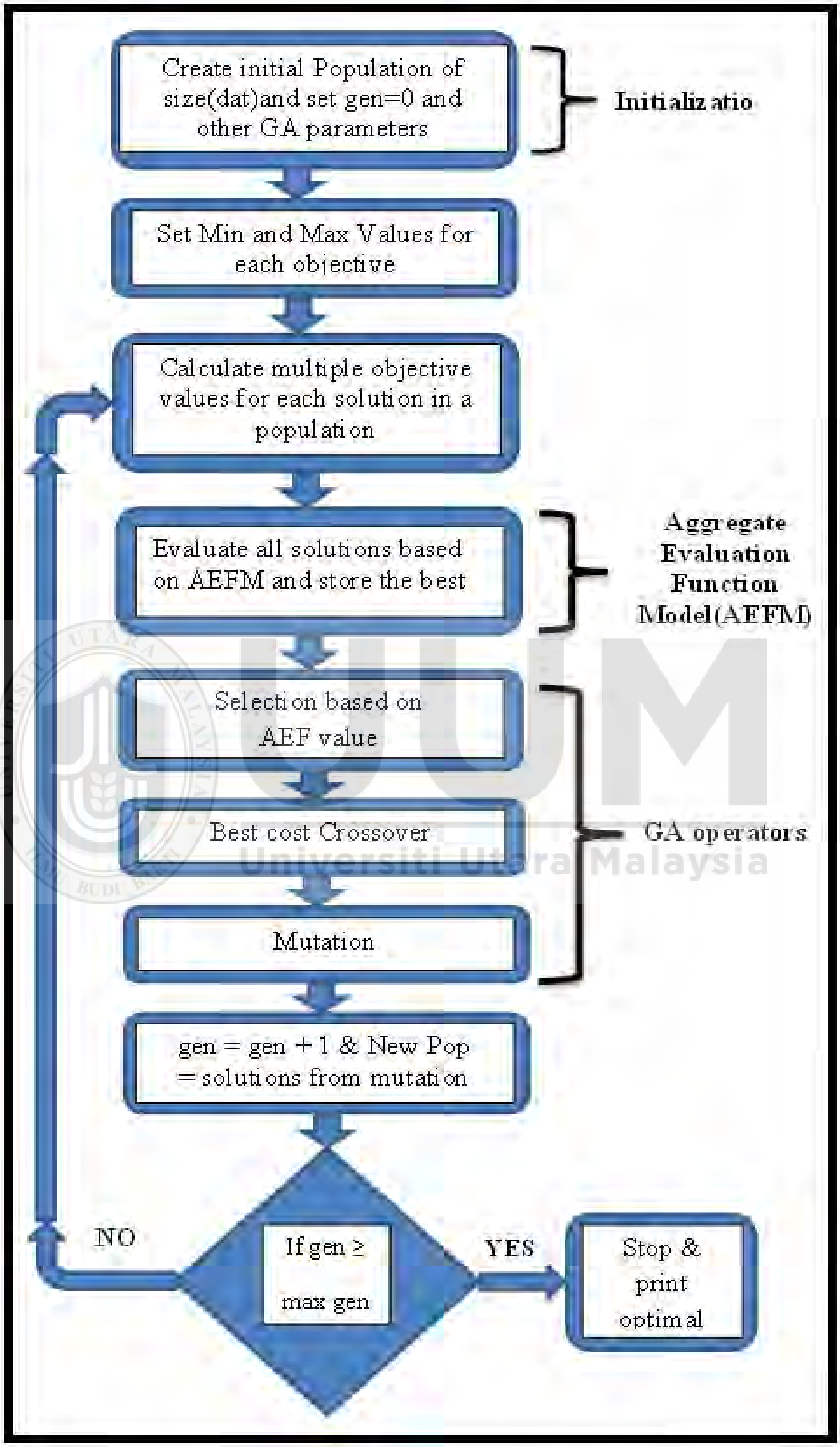


Figure 3.11. Flowchart optimization methodology

3.5 Summary and discussion

In order to test the system in correct manner, must be the input signal consideration. Input signals must sufficiently excite or probe the system to ensure that the measured outputs completely characterize the nonlinear system model. Generally, the input must be sufficiently rich in both values amount content and amplitude variation. It is important tests for nonlinearity. Is the system nonlinear or not? If so (nonlinear), can the type of nonlinearity (for example, polynomial). As well as, model class selection, based on the type and severity of the nonlinearity, one can choose between various parametric and nonparametric model classes. As a general rule, whenever parametric models provide adequate descriptions of the system, they are preferable to nonparametric models because they are more robust to errors and admit relatively simple identification procedures. Also must be considered for Optimality criterion selection. Once a given model class is adopted, a criterion or cost function is used to assess the fitness of specific models within the chosen class. System identification is the process of selecting or estimating the model that optimizes the criterion. The most common criterion is the sum of squared errors between the predicted output produced by the model and the actual measured Output. These steps are important so that the newly formulated algorithms can produce control model that could help human to understanding and overcoming the conflicting problem better, especially for large dimension problems.

CHAPTER FOUR
CONSTRUCTION AND OPTIMIZATION OF NON-WEIGHTED AGGREGATE EVALUATION FUNCTION

4.1 Introduction

In this chapter, demonstrate how to construct and optimize the aggregate evaluation function (AEF) developed based on the methodology presented in the previous chapter. Section 4.1 illustrate introduction. Section 4.2 presents the method of data collection and how to read that data and the access sources to it. As well as how to deal with the factors that affect on knocking. Section 4.3 provides how to construct the individual objective. Aggregate multi-objective evaluation function constructions demonstrate in Section 4.4. The optimization process to the factors was explains in Section 4.5. Finally, section 4.6 presents the validation of the model.

Universiti Utara Malaysia

4.2 Data Selection and Reading

The development of the evaluation function model was based on the gathered data. Most multi-objective optimization problems can collect data at different times. In other words, the data for a particular objective can be collected in a given day and data for another objective the next day. Following the nature of our problem, the data was collected for all objectives simultaneously from multi-sensors. This kind of data is known as "overlapping data" and this technique is called “multi-sensors data fusion”. A diagnostic tool known as OBD II scan tool that can read the data from the sensors located in the engine of the vehicle was used, see Figure 4.1.

Figure 4.1. Engine Diagnostic tool ULTRASCAN P1 (OBD II scan tool)

Universiti Utara Malaysia

In automotive field, “knocking” is a nonlinear multi-objective optimization problem in internal combustion engine, whereas the factors that cause this problem are non-linear and conflicting simultaneously. Furthermore, mutual influence exists between those factors (Harishchandra, Ravindra, Sachin, & Thiyagarajan, 2010; Merola et al., 2011; Zhen et al., 2012). This device addresses the electronic control unit (ECU) to read the real values of the factors that affect the occurrence of knocking, such as mechanical, electrical, environmental, and misuse. Figure 4.2 shows the fishbone diagram used to clarify those factors that influence knocking.

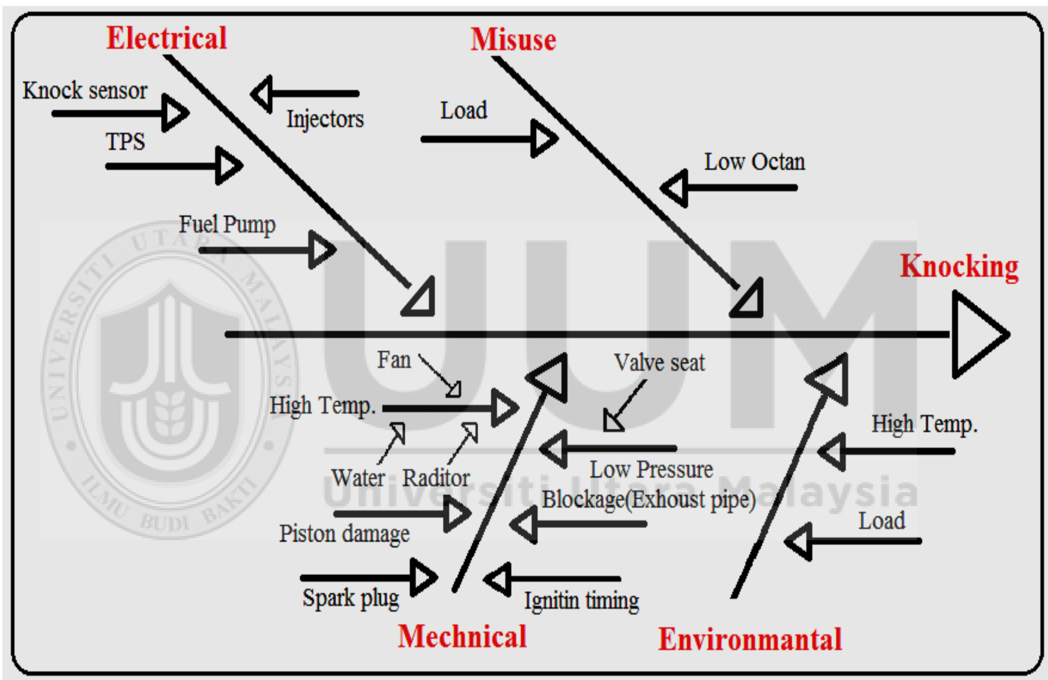


Figure 4.2. Fishbone Diagram (Knock problem)

Real data obtained from PROTON and EGMA companies for different engines and in different conditions. The three major factors considered were temperature (TEPM), throttle position sensor (TPS), and the speed engine (RPM). In addition, the output data from Knock sensor were obtained to build the mathematical model.

Several groups of real data (data sets) were obtained. These groups were divided into two parts, namely, training and testing data. Training data were used to construct the proposed model for function evaluation. Training data were obtained from a test engine (1.3 L Campro, modified to turbocharger, four-cylinder, MPI). Testing (experimental) data were used to validate the model, and these data were obtained from different engine types and conditions, can see the data in Appendix.

The proposed model has been applied to many real data obtained from various internal combustion engines. Been dealing with these engines of different conditions such as temperature, speed and load specifications and some of them can be summarized in the following Table 4.1.

Table 4.1

Different test Engines and Conditions

Name	Load	Speed (RPM)	Outside Temperature
Proton Turbo-Charge	With load	1000-5000	33 C^0
Jeep-Cherokee 6V- 3600 cc	With load	760-2620	15 C^0
Hyundai-Genesis 8V 5.0GDI	With load	540-3156	19 C^0
Dodeg	With load	740-2600	25C^0
KIA-Sportage 4cyl-G2.0cc DOHC	With load	773-4164	17 C^0
KIA-Sorento 4cyl- G2.4cc DOHC	With load	665-3473	18 C^0

Universiti Utara Malaysia

4.3 Construct Individual Objectives and Aggregate Multi-objective Evaluation Function

After collecting the data, the concept of System Identification (SI) was applied in the construction of individual objective function. SI is a link between the real world and the world of mathematics. This link was between observation data (input/output data) and mathematical model, representing the behavior of these data. This relationship can be represented by a mathematical model, graph, or table. SI needs a cost function (CF), which is calculated to obtain the difference between the data and the model. The minimized variation maximizes the accuracy in the representation.

In our problem, "curve fitting" was used to obtain the optimal model for each objective. We also used a statistical technique, "Akaike's Information Criterion" (AIC), as a cost function to evaluate the mathematical model that represents the data.

The mathematical relationship can be determined using curve fitting. Curve fitting seeks to identify an appropriate curve in which to fit the observed values and then utilizes the curve function to analyse the relationship between variables.

Curve fit is able to express the relationship between the dependent variable Y and the single independent variable X and therefore estimate the value of parameters using nonlinear regression in order to determine an appropriate mathematical model.

The purpose of curve fitting is to determine a mathematical model that suitable fits the data. It is assumed that the selection of a function of a certain form will have a theoretical basis and therefore the curve fit will determine the specific coefficient that the function matches with the data (Pro, 2007).

4.3.1 Curve Fitting Technique

Curve fitting (CF) is used to discover a function f(x) in a function class M for the data (x_i, y_i) where i=0, 1, 2,…, n–1. The function f(x) will minimize the residual (error) under the weight W. The residual refers to the distance between the data samples and f(x). A smaller residual suggests a better fit. In the context of geometry, curve fitting refers to a curve y = f(x) that fits the data (x_i, y_i) where i=0, 1, 2,…, n–1.

The fitting model should be selected before fitting the data set. One choice is to use a non-linear model to fit the logarithmic data, which leads to a correct fitting result. However, an improper choice such as a linear model results in incorrect fitting or inaccurate characteristics of the data set. Therefore, an appropriate fitting model must be selected based on the data distribution shape, and then based on the result, determine if the model is suitable.

Universiti Utara Malaysia

4.3.2 Curve Fitting Methods

Differing fitting methods examine the input data in order to determine the curve fitting model parameters. Every method has a unique set of criteria to evaluate the fitting residual in order to determine the fitted curve. When the criteria of each method is properly understood, the best method can be determined and applied to the data set and fit the curve. The least squares (LS), least absolute residual (LAR), Bi-square fitting method to linear fit, power fit, exponential fit, Gaussian peak fit, logarithm fit as well as others can be applied in order to find the function f(x).

LS method determines f(x) by minimizing the residual based on the following formula:

$$\frac{1}{n}\sum_{i=0}^{n-1} w_i(f(x_i) - y_i)^2 \tag{4.1}$$

where n refer to the number of data samples, w_i is i^{th} element of the array of weights for the data samples, $f(x_i)$ is i^{th} element of the array of y-values of the fitted model, and y_i is i^{th} element of the data set (x_i, y_i).

LAR method determines f(x) by minimizing the residual based utilizing the formula:

$$\frac{1}{n}\sum_{i=0}^{n-1} w_i|f(x_i) - y_i| \tag{4.2}$$

Bisquare method determines f(x) utilizing an iterative process, displayed in Figure 4.3. The residual is also calculated utilizing the same formula as the LS method. Bisquare method calculates the data beginning with iteration k.

Universiti Utara Malaysia

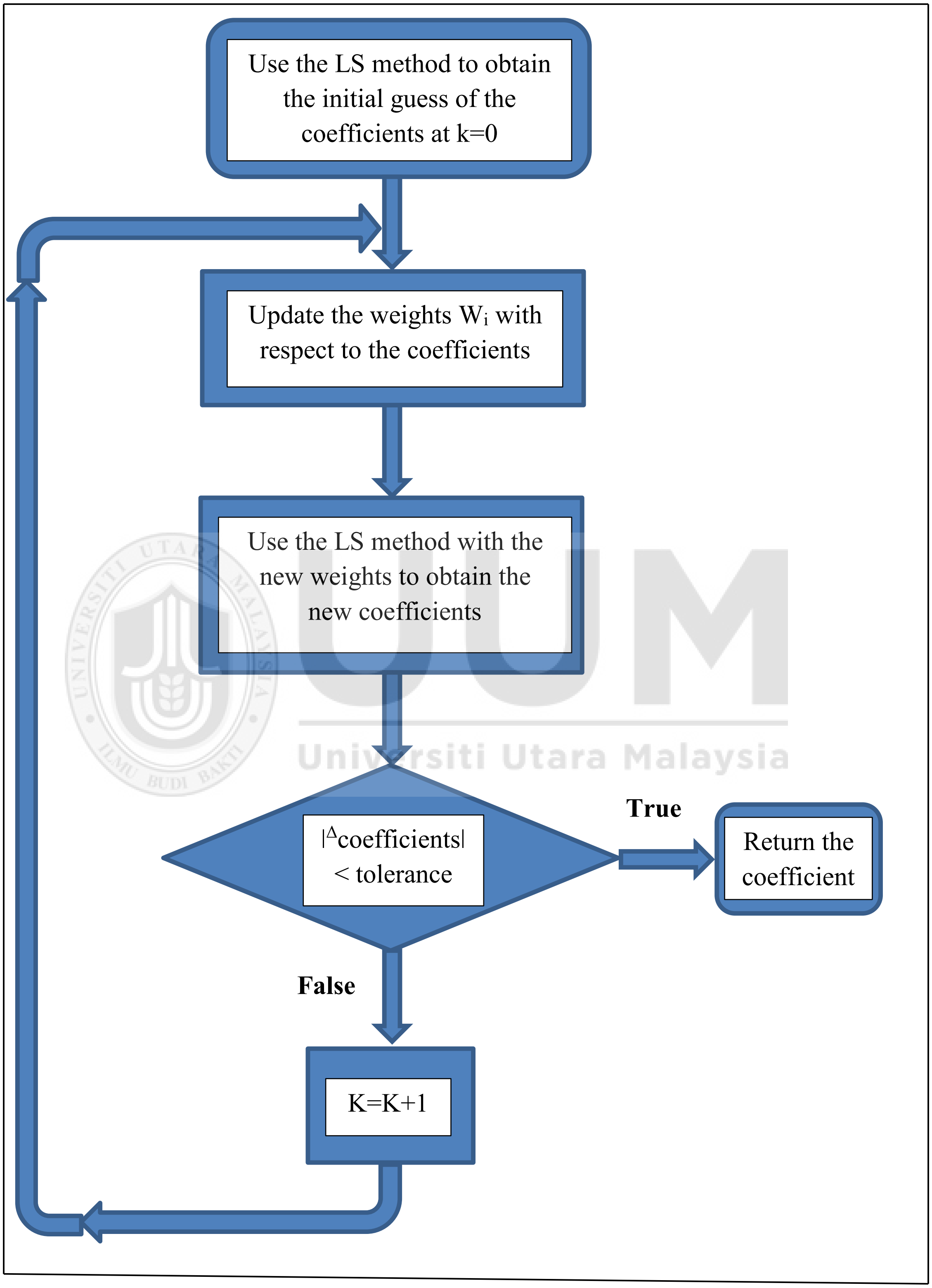


Figure 4.3. Bisquare Method Flowchart

Three factors (TPS, RPM, TEMP) were influential on the knocking problem. Each factor was considered an objective, represented as a mathematical model. A non-linear relationship was found between each objective and knocking. Figure 4.4 shows the non-linear relationship between (TPS) and knock.

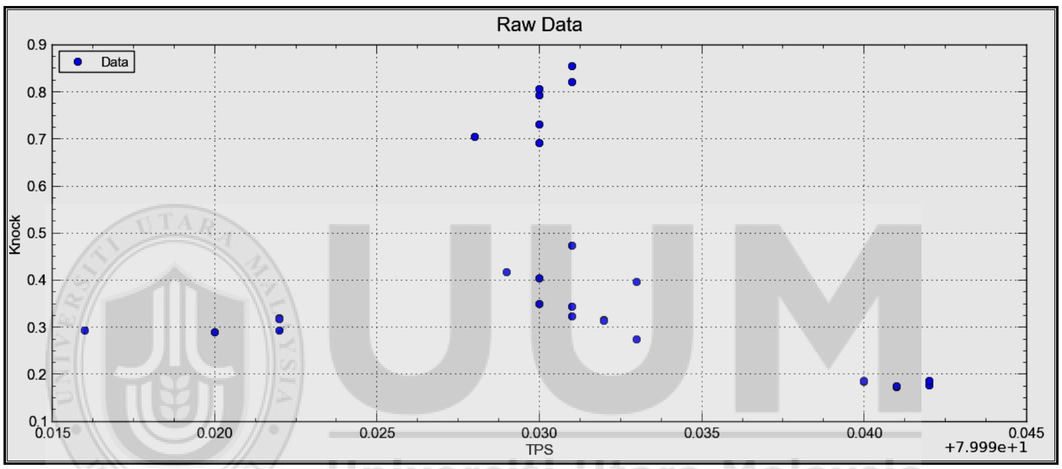


Figure 4.4. Nonlinear Relationship between TPS and Knock

Therefore, must identify and apply the models that fit with the nature and behaviour of non-linear data for each objective. Many classes of models (e.g., distribution models, exponential models, sigmoidal models, growth models, decline models, dose response models, etc.) for non-linear data set were selected to obtain the optimal model suitable for the data set.

Approximately 50 models were applied for each objective (factor), as shown in Figure 4.5. Based on some statistical measurements (e.g., AIC, residuals, R, R2 , std. err), the best three models(top results) were obtained and therefore selected one of them as an optimal model to represent a data set.

Name	Kind
Sinusoidal	Regression
Gaussian Model	Regression
Heat Capacity	Regression
Steinhart-Hart Equation	Regression
Reciprocal Quadratic	Regression
Logistic Power	Regression
Ratkowsky Model	Regression
Rational Model	Regression
Rational Model	Regression
Natural Logarithm	Regression
Wavy	Regression
Normal (Gaussian) PDF	Regression
Saturation Growth Rate	Regression
Reciprocal	Regression
Piecewise Linear	Regression
Bleasdale	Regression
Logistic	Regression
Harmonic Decline	Regression
Exponential Association 2	Regression
Log Normal CDF	Regression
Normal (Gaussian) CDF	Regression
Harmonic Decline	Regression
DR-LogProbit	Regression
DR-Hill-Zerobackground	Regression
DR-Hill	Regression
MMF	Regression
Richards	Regression
Farazdaghi-Harris	Regression
Reciprocal Logarithm	Regression
Reciprocal Logarithm	Regression
Vapor Pressure Model	Regression
Log Normal PDF	Regression
DR-LogProbit-Zerobackground	Regression
DR-Multistage-1-Zerobackground	Regression
DR-Logistic-Zerobackground	Regression
DR-Multistage-2-Zerobackground	Regression
DR-Probit-Zerobackground	Regression
DR-Weibull-Zerobackground	Regression
DR-Gamma-Zerobackground	Regression
DR-LogLogistic-Zerobackground	Regression
DR-Multistage-1	Regression
DR-LogLogistic	Regression
DR-Multistage-3-Zerobackground	Regression
DR-Gamma	Regression
DR-Logistic	Regression
DR-Multistage-2	Regression
DR-Probit	Regression
DR-Weibull	Regression
DR-Multistage-4-Zerobackground	Regression
DR-Multistage-3	Regression
DR-Multistage-4	Regression

Universiti Utara Malaysia

Figure 4.5. Non-linear Regression Models

Obtained the best three models (Sinusoidal, Gaussian model, Heat capacity) as the top results to represent the objective (TPS) (see Figure 4.6). Following AIC, R, R^2, and std. err, we selected the Sinusoidal model as a best model to represent the data for this objective.

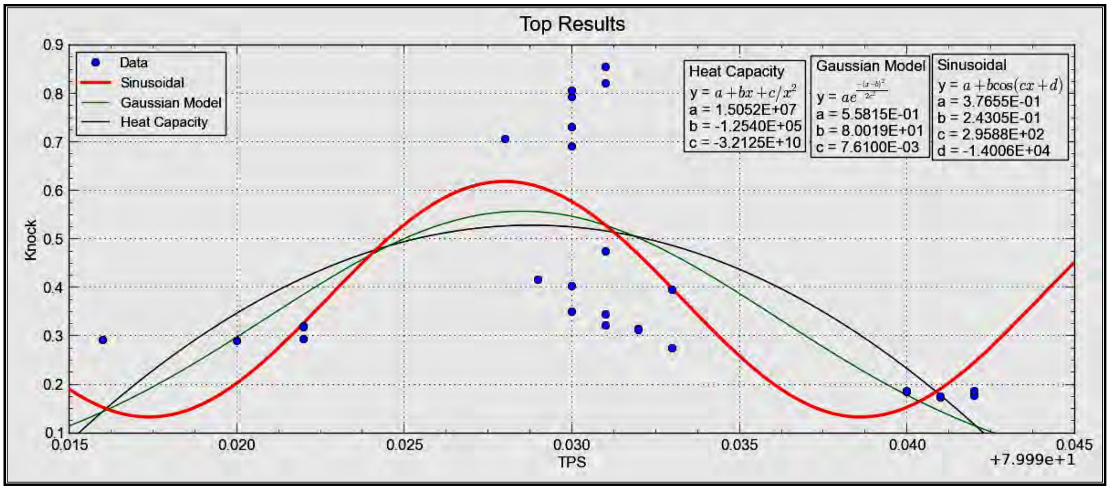


Figure 4.6. Best Three Curve Fitting Models for TPS

Based on the results of AIC, R, R^2, and std. err, we selected the Sinusoidal model, as shown in Figure 4.7, as the best model to represent the data fit for this objective.

UUM Universiti Utara Malaysia

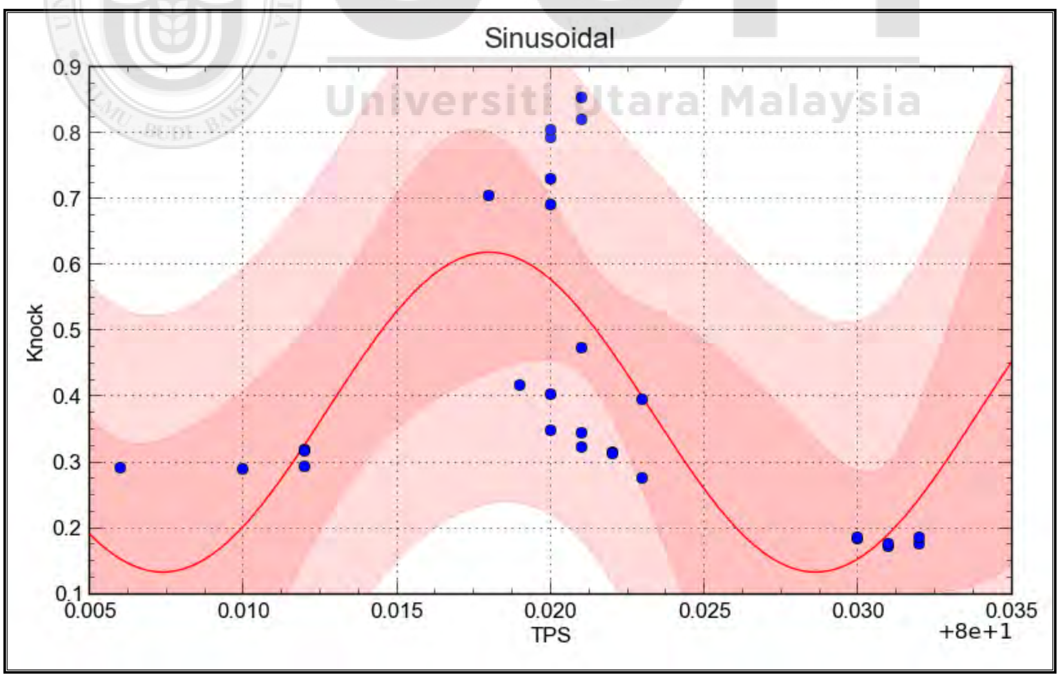


Figure 4.7. Sinusoidal Curve Fitting for TPS

4.3.3 Akaike Information Criterion

AIC is a method to select a model from a set of models. AIC examines the trade-off that exists between the complexity and the suitability of fit for the model.

AIC is derived from relative Kullback-Leibler information (K-L). When an approximating model is used to approximate the full reality, information can be lost. A 'good' model is one that is close to *f (f* truth in terms of a probability distribution) with a smaller relative K-L information value. The AIC value for each model can be calculated with the same data set, where the 'best' model refers to the one with minimum AIC value (S. Hu, 2007).

4.3.4 Information Loss Estimation by Akaike Information Criterion

A model was selected based on the values that were obtained from (AIC). Information will be lost as only one candidate model is used to represent the 'true' model. From N candidate models, the model that causes the least loss of information can be selected.

AIC candidate model values are denoted by AIC_1, AIC_2, AIC_3, …, AIC_N. AIC_{min} is considered the minimum of those values. Therefore, $\exp((AIC_{min}-AIC_i)/2)$ is interpreted as the relative likelihood that i^{th} model minimizes the (estimated) information loss (Aho, Derryberry, & Peterson, 2014).

For example, candidate set has three models with AIC values of 200, 202, and 210. The second model is exp((200-202)/2) = 0.3679 times as likely as the first model to minimize information loss and the third model is exp((200-210)/2) =0.0067 times as likely as the first model to minimize the information loss. The third model is excluded. The weighted average of the first two models is obtained (with weights 1 and 0.368) and a statistical inference conducted based on the weighted multi-model (Akaike, 1998). An alternative method is to obtain further data to distinguish the first two models.

If represent $((AIC_{min}-AIC_i)/2)$ as b, can express a relative likelihood A as:

$A= e^b$, where e = 2.7182

Now can calculate the value of *A* as vary *b*. Values obtained in this manner is listed in Table 4.2. For example,

$A= \exp((AIC_{min} – AIC_i)/2)$, $AIC_{min} = 2$, $AIC_i = 6$, $b = ((2-6))/2) = -2$, $e = 2.7182$

$A= e^{-2}= 2.718^{-2} = 0.135$

Universiti Utara Malaysia

Table 4.2

Value of A as Vary b

b	**-3**	**-2**	**-1**	**0**	**1**	**2**	**3**
$A=e^b$	0.050	0.135	0.368	1	2.718	7.389	20.086

Can also represent the variances of b in a graph, as shown in Figure 4.8

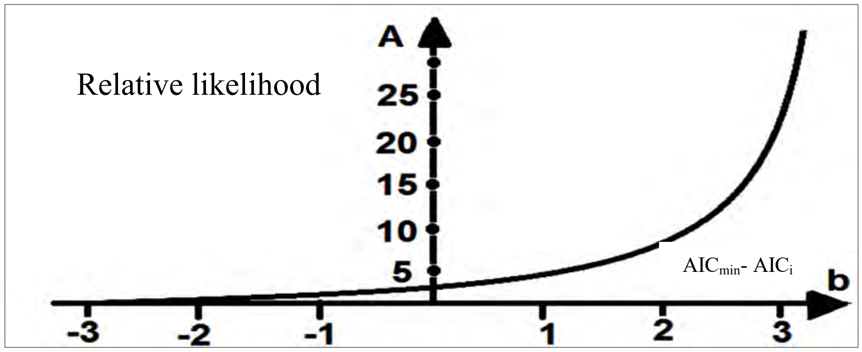


Figure 4.8. Graph of the Exponential Function A = e^b

4.3.5 Goodness of Fit of a Model

Goodness of fit (GOF) of a model is generally assessed using the coefficient of determination, R^2. The assessment method equals one minus the ratio of the sum of squared estimated errors (the deviation of the actual value of the dependent variable from the regression line) to the sum of squared deviations of the means of the dependent variable. The total variation of the dependent variable is the sum of squared deviations of its mean. The extent to which the regression fails to explain the dependent variable (a measure of noise) is a measure from the sum of squared deviations of the regression line. Therefore, R^2 statistic measures the extent to which the total variation of the dependent variable is explained by the regression. A variation in the dependent variable is well explained by a higher value of R^2, which is an important factor if the model is to be used for predicting and forecasting (Sykes, 1993).

4.3.6 Residuals

GOF is determined within an iterative cyclic process. To ensure the fitting process is the most appropriate for the data set, the curve fitting algorithm associated with the model is measured based on the cycle when the curve is modified. Curve fitting algorithms are designed to minimize the sum of the residuals squared and GOF can be determined measuring the residuals in every cycle. Figure 4.9 shows the data are distributed randomly, indicating that this curve fitting is not unlikely.

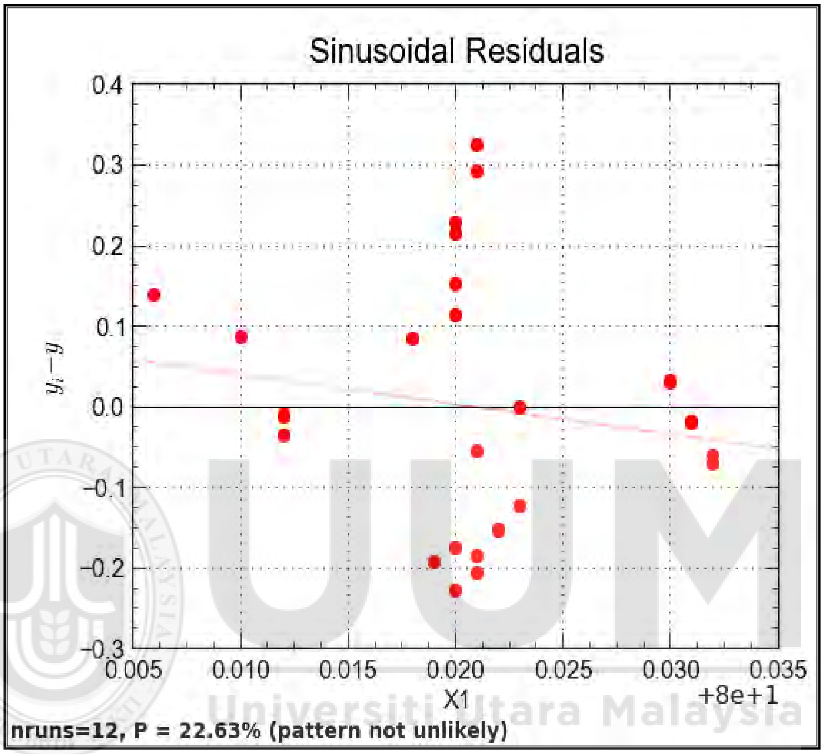


Figure 4.9. GOF Measure

The residuals allowed the difference between the result and the data as a function of the independent variable to be revealed. A straight-line fit among the residual points was also apparent in the residual plot. The light red regression line outlined the trend of the data and whether it was upward or downward (shown by the slope) as well the residual bias, either upwards or downwards.

The Wald-Wolfowitz test was also conducted on the residuals. The bottom of the plot lists the observed quantity of runs as well as the chances that the observed quantity of runs would occur if the model used to fit the data is correct; that is, the residuals being randomly distributed around the curve. The pattern of residuals is unlikely if the probability is less than 5%. If the pattern is greater than 5%, then the pattern is not unlikely. It is more desirable to obtain a higher probability of likelihood.

4.3.7 Estimation of Regression Model

One of the common approaches used to estimate the curved fitting for the regression model is least squares error approach (LSE) (Joaquim & Marques, 2007). This approach attempts to obtain the minimum amount of error between observed value y_i and estimation value $b_0 + b_1x_i$:

$$E = \sum_{i=1}^{n} \varepsilon_i^2 = \sum_{i=1}^{n} (y_i - b_0 - b_1 x_i)^2, \quad (4.3)$$

where b_0 and b_1 are estimates of β_0 and β_1, respectively.

Other statistical measures like R, R^2 and adjusted R^2 are used in curve fitting to estimate the model. These measures are useful to assess GOF of the model. By computing the error sum of squares or residual sum of squares (SSE), that is, the quantity E in Eq (4.3), can understand their meaning.

The predicted (or fitted) value is $\hat{y}_i = b_0 + b_1x_i$.

The computed error $e_i = y_i - \hat{y}_i = y_i - b_0 - b_1x_i$ is known as the residuals (Cottrell, 2003).

$$SSE = \sum (y_i - \hat{y}_i)^2 = \sum e_i^2 \quad (4.4)$$

The deviations are referred to each predicted value. Therefore, SSE has $n - 2$ degrees of freedom (DOF) because two DOF are lost: b_0 and b_1.

Mean square error: $\text{MSE}= \frac{\text{SSE}}{n-2}$ (4.5)

Root mean square error or ***standard error***: RMS $=\sqrt{MSE}$.

The total variance of the observed values is related to the total sum of squares (SST):

$\text{SST}= \sum (y - \bar{y})^2$ (4.6)

The contribution of X to the prediction of Y can be evaluated using the following association measure known as coefficient of determination or R-square (Cottrell, 2003).

$R^2= \frac{\text{SST}-\text{SSE}}{\text{SST}} \in [0,1]$ (4.7)

Often the value of "R-square" is found to be slightly optimistic. Several authors proposed the use of adjusted R-square instead.

$R_a^2 = R^2 - (1 - R^2)/(n - 2)$ (4.8)

4.3.8 Comparison between (Sinusoidal and Gaussian model) in TPS objective

The best model is obtained from the results in Table 4.3 based on some tests.

Universiti Utara Malaysia

Table 4.3

Overview (Sinusoidal and Gaussian model) in TPS objective

	Sum of Square	**R^2**	**DOF**	**AICc**
Sinusoidal	0.633329	0.537640	24	-99.0912
Gaussian Model	0.717412	0.476255	25	-98.1207

- **Comparison**

By testing (AICc), the model (Sinusoidal AICc = -99.09) is the closest model to represent the data (i.e, best regression). By observing the value of probability (61.8987%), the probability of reducing the loss of information is highest for the Gaussian model with a probability of 38.1013%.

The best regression selected using the F test is the Gaussian Model.

- **Justification**
  - **AIC Criterion**

AIC for Sinusoidal (-99.0912) was better than the Gaussian Model (-98.1207).

Delta = 0.970501

probability = 0.381013

The Sinusoidal model was better at61.8987%

  - **F-Test**

The more complicated model (Sinusoidal) has better SST than the simpler model (Gaussian model). The result of an F-test is strictly not valid unless the simpler model is a subset of the more complex model; that is, the simpler model can be obtained by setting some parameters in the more complex model to certain constants.

F = 3.18634

P = 0.0869025

If the simpler model was correct, SST would increase by approximately the gain in DOF when moving from the complex model to the simpler one (i.e., F = 1).

The simpler regression (Gaussian model) has 8.69025% probability of better fit to the data.

R^2 is a number between zero and one Cottrell (2003), and this value shows the percentage change in knocking, simply indicating the extent to clarify regression analysis of the value of the variable that we are attempting to predict knocking. In this case, the value is equal to R^2 0.537, implying that the regression analysis model is strong and can calculate the value of knocking approaches healthy status. If the value of this number is 0.3, should neglect the equation and search again for the factors that affect knocking. Graphically, can illustrate this comparison in Figure 4.10.

Universiti Utara Malaysia

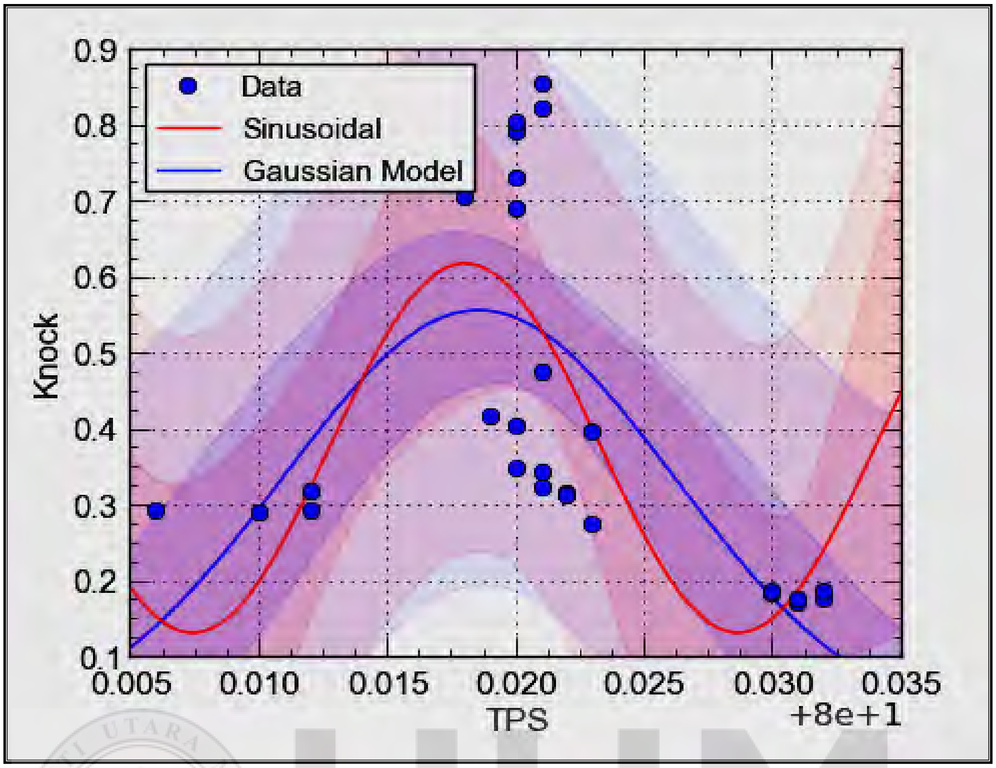


Figure 4.10. Compare between Two Models (Sinusoidal vs Gaussian Model)

Finally, Sinusoidal was obtained as the best regression model and expressed as

Y = a + b*cos(c*X + d) (4.9)

4.4 Objectives Aggregation

All these procedures were applied on one objective (TPS factor) to obtain a mathematical model that simulates the behavior of knocking. The same procedure was carried out on other objectives (factors) (RPM, TEMP). Hence, three models with the following equations was attended:

Knock = a + b*cos(c*TPS + d) (4.10) (Sinusoidal)

knock= p + q*Rpm+ s*Rpm^2 (4.11) (Polynomial second order)

knock= m + n *cos(o*Temp + e) (4.12) (sinusoidal)

These factors (TPS, RPM, TEMP) have associated partial effects (partial influence) on knocking. To determine the overall effect on knocking should find the sum of these effects. In this study, the partial effect of each factor affects the process of knocking, and thus the sum of these effects on those factors to create the overall effect of the factors on knocking was found. A better model was obtained for one factor that affects knocking. Then, the first derivative of the model was found to determine the partial effect of the factor after applying real data, considering the fixed values of the other factors.

First, aggregate all three individual objectives into a single objective. To calculate the partial effect for each factor, a partial derivative was used. The proposed model comprised three factors (complex model), Hence, the model analysis can be complicated. The first partial derivative was used to simplify the model to determine a partial effect for each factor. For example, the first partial differentiation of function Z = f(x,y) bivariate (two dimensions) provided a tangent line at a given point direction x or y, and thus the rate of change in Z with respect to x or change rate for Z with respect to y was measured. The first partial derivative was calculated using some differentiation rules, as shown below.

$knock = a + b * cos(c*Tps + d)$, $\quad dK = \left(\frac{\partial k}{\partial Tps} dTps\right)_{Rpm,Temp}$,

where k denotes the knock. The derivative of knock (k) with respect to TPS, holding RPM, TEMP as constants, is known as the partial derivative of knock with respect to TPS.

Five rules was needed to derive the first partial derivative of this formula:

$\frac{d}{dx}c = 0$ ……………rule (1)

$\frac{d}{dx}cx = c$ …………….rule (2)

$\frac{d}{dx}cu = c\frac{du}{dx}$ ……..………rule (3)

$\frac{d}{dx}\cos x = -\sin x$ ……..……….rule (4)

If $y = cos(u)$ such that $u=f(x)$, then

$$\frac{dy}{dx} = -\sin(u).\frac{du}{dx}$$ ……..……….rule (5)

Therefore,

$a = 0$, constant…….rule (1)

$b*u=b*cos(c*TPS+d)$

because $c*Tps = c$…………...rule (2), $d = 0$........……rule (1)

if $y = cos(u)$ such that $u=f(x)$, then

$$\frac{dy}{dx} = -\sin(u).\frac{du}{dx}$$

$\frac{du}{dTps}=\frac{d}{dTps}cos(c*Tps+d)= -sin(c*Tps+d)*c$ ……..……rule (5)

$\frac{d}{dTps}bu=b*\frac{du}{dTps}= - b*c\ sin(c*Tps + d)$………….rule (3)

$$\boldsymbol{\frac{\partial k}{\partial Tps} = -b * c * \sin(c * Tps + d)}$$

For RPM factor,

$knock = p + q*Rpm+ s*Rpm^2, dk = \left(\frac{\partial k}{\partial Rpm}dRpm\right)_{Tps,Temp}$,

Universiti Utara Malaysia

$$\frac{\partial \boldsymbol{k}}{\partial \boldsymbol{Rpm}} = \boldsymbol{q} + \mathbf{2} * \boldsymbol{s} * \boldsymbol{Rpm}$$

For Temp factor,

*knock = m + n *cos(o*Temp + e)*, $dk = \left(\frac{\partial k}{\partial Temp} dTemp\right)_{Rpm,Tps}$,

$$\frac{\partial \boldsymbol{k}}{\partial \boldsymbol{Temp}} = -\boldsymbol{n} * \boldsymbol{o} * \boldsymbol{sin}(\boldsymbol{o} * \boldsymbol{Temp} + \boldsymbol{e})$$

Now, an overall model was obtained, which can be used in the analysis of the effect of factors on knocking, given that knocking depends upon RPM, TEMP, and TPS. Thus, the following formula was reached:

k = k(Rpm,Temp,Tps)

To obtain the relation among RPM, TEMP, and TPS, a partial deferential was used:

$$dk = \left(\frac{\partial k}{\partial Rpm} dRpm\right)_{Tps,Temp} + \left(\frac{\partial k}{\partial Temp} dTemp\right)_{Rpm,Tps} + \left(\frac{\partial k}{\partial Tps} dTps\right)_{Rpm,Temp} \quad (4.13)$$

Substituting the value of (4.10),(4.11),(4.12) in Eq. (4.13), was obtained

$$k = (q + 2 * s * Rpm)_{Temp,Tps} dRpm + (-n * o * sin(o * Temp + e))_{Rpm,Tps} dTemp + (-b * c * \sin(c * Tps + d))_{Rpm,Temp} dTps \quad (4.14)$$

Knock =-1*(0.24304*20.873*sin((29.873*tps-1405.39002)*pi/180)-1.08452*0.000117*rpm-0.000009226 * 38.29344 * sin((13.29344* temp+35.15755)*pi/180)+5)

4.5 Multi-Objectives Optimization using continuous Genetic Algorithm

One way to solve the problems of multi-objective optimization is to use aggregate multi-objectives in a single objective. To determine the weights for each objective is difficult because they are characterized by conflicting and mutual influence

To aggregate the several objectives models into a single objective overcomes the problem of determining the weights and partial mutual influence. This model has been used as a scalar function of the continuous genetic algorithm (GA).

In such a problem, each variable requires many bits to represent it. If the number of variables is large, the size of the chromosome is also large. Continuous GA is selected to achieve the final goal of increasing the accuracy of the results. When the variables are continuous, to represent them by floating-point numbers is more logical. A single floating-point number represents the variable instead of N_{bits} integers, requiring less storage than the binary GA. In continuous GA, the chromosomes do not have to be decoded prior to the evaluation of the scalar function. Therefore, the continuous GA is inherently faster to obtain than the binary GA.

To solve for some of the multi-objective optimization problems, can be search for the optimal solution to the problem through the problem variables. Therefore, begin the process of fitting the solution to GA by defining a chromosome as an array of variable values to be optimized. 3D optimization problem was given and represented as the chromosome (TPS, RPM, TEMP). Then, the chromosome is written as an array with 1* 3 elements such that

Chromosome = (TPS, RPM, TEMP)

Each chromosome has a scalar determined by evaluating the scalar function *f* at the variables, TPS, RPM, TEMP.

Scalar = *f* (Chromosome)=*f* (TPS, RPM, TEMP)

Scalar =*f* (TPS, RPM, TEMP) = -1*(0.24304*20.873*sin((29.873*tps-1405.39002)* pi/180) -1.08452* 0.000117*rpm -0.000009226 * 38.29344*sin((13.29344*temp+35.15755)*pi/180) +5)

Subject to the constraints

$$\boldsymbol{min \leq\ Tps \leq max}\ ,$$

$$\boldsymbol{min \leq\ Rpm \leq max \quad and}$$

$$\boldsymbol{min \leq\ Temp \leq max}$$

For example:

$$740 \leq Rpm \leq 2600$$

$$2.500 \leq Tps \leq 12$$

$$50 \leq Temp \leq 68$$

4.5.1 Initial Population

To begin the continuous GA, was create an initial population of size (dat) and set generation = 0. For other GA parameters, we define an initial population of N_{pop} chromosomes. A matrix represents the population with each row in the matrix 1 * N_{var} array (chromosome) of continuous values. Given an initial population of N_{pop}

chromosomes, the full matrix of N_{pop} * N_{var} real values is obtained by reading data set as a file.

dat=xlsread('PROTON_4FACTORS_CY1.xlsx')

In our example, set the minimum and maximum values of each objective.

varhi1=12	max tps
varlo1=2.5000	min tps
varhi2=2600	max rpm
varlo2=740	min rpm
varhi3=68	max temp
varlo3=50	min temp

These values vary from one engine to another depending on the design.

The multiple objective values for each solution in a population was calculated, to obtain the overall assessment of each solution in the population (called scalar function).

Scalar = *f* (Chromosome) = *f* (TPS, RPM, TEMP)

4.5.2 Natural Selection

By selection operator and according to the evaluation function values, we decide which chromosome in the initial population is fit enough to survive and possibly reproduce offspring in the next generation while the rest die off. This process is conducted at each iteration to allow the population of chromosomes to evolve over the generations to the fittest members as defined by the evaluation function. In this process, N_{pop} scalar values and associated chromosomes are ranked from the lowest to the highest value. In N_{pop} chromosomes in a given generation, only N_{keep} chromosomes were retained and deemed fit enough for mating. Fraction of N_{pop} that survived for the next step of mating was determined by rate, X_{rate}. The number of chromosomes that were at each generation was

$$\mathbf{N_{keep} = X_{rate} * N_{pop}}$$

Natural selection occurs in each iteration or generation of the algorithm. Among N_{pop} chromosomes in a generation, only the top N_{keep} survived for mating, and the number of discarded chromosomes (bottom fraction) was determined by N_{pop}-N_{keep} to accommodate the new offspring. Based on random choice or somewhat arbitrary, we decide on how many chromosomes are retained. The few chromosomes retained to survive the next generation limited the diversity of genes in the offspring, and to retain too many chromosomes provide them the opportunity to contribute their traits to the next generation. The natural selection process often retains 50% (X_{rate} = 0.5). If we have N_{pop}= 60, therefore

$$N_{keep}=N_{pop}*N_{rate}$$

$$N_{keep}=60*0.5= 30$$

"Thresholding" method is another approach to natural selection. All chromosomes that have lower cost than some thresholds survive.

4.5.3 Pairing Approaches

In GA, pairing chromosomes can be useful and interesting, such that they varied as pairing in an animal species. We select two chromosomes from a pool of N_{keep} chromosomes to produce two new offspring. Pairing is conducted in the mating population until N_{pop} - N_{keep} offspring are born to replace the discarded chromosomes. Pairing has several methods.

1. **Pairing from Top to Bottom.** This approach works on N_{pop} matrix, starting from the top of list in the selection of chromosomes to include in the mating process. The approach selects two chromosomes simultaneously until the top N_{keep} chromosomes are selected for mating. Therefore, the algorithm pairs odd rows with even rows. If a certain population matrix is present, the mother has row numbers in the population matrix, ma= 1, 3, 5,......., and the father has the row numbers pa = 2, 4, 6… This approach does not accurately model nature, but the programming is much more simple (Haupt & Haupt, 2004).

2. **Random Pairing.** Uniform random number generator is used in this approach to select chromosomes. The parent's row numbers are found utilizing the following equation:

   ma = ceil (N_{keep}*rand(1, N_{keep}))

   pa = ceil (N_{keep}*rand(1, N_{keep}))

   where ceil rounds the value to the next highest integer.

Universiti Utara Malaysia

3. In this study, **weighted random pairing** is utilized. The probability which is assigned to each chromosome in the mating pool is inversely proportional to its cost. A chromosome with a lower cost has a greater chance of mating, while a chromosome with a higher cost has a lower chance of mating. A random number determines which chromosome is selected. This is often known as roulette wheel weighting. Rank and cost are two techniques of weighting.

a. **Rank weighting.** This approach is based on a probabilistic calculation through rank n for each chromosome, which is an independent problem.

Suppose there is an initial population of 10 random chromosomes (N_{pop}=10). Their corresponding cost is shown in Table 4.4.

Table 4.4

Initial Population of 10 Random Chromosomes (N_{pop}=10) and Their Corresponding Cost

X	Y	Cost
6.9745	0.8342	3.4766
0.3035	9.6828	5.5408
2.4021	9.3359	-2.2528
0.1875	8.9371	-8.0108
2.6974	6.2647	-2.8957
5.6132	0.1289	-2.4601
7.7246	5.5655	-9.8884
6.8537	9.8784	13.752
6.5454	5.3938	12.627
4.5874	7.4206	-1.7615

Therefore, (N_{keep}= 5) surviving chromosomes after a 50% selection rate was obtained (see Table 4.5).

N_{keep}=10*0.5=5.

Table 4.5

Surviving Chromosomes after a 50% Selection Rate

Number	X	Y	Cost
1	7.7246	5.5655	-9.8884
2	0.1876	8.9371	-8.0108
3	2.6974	6.2647	-2.8957
4	5.6130	0.1288	-2.4601
5	2.4021	9.3359	-2.2528

Based on the formula,

$$P_n = \frac{N_{keep} - n + 1}{\sum_{n=1}^{N_{keep}} n}$$

The probability for each chromosome can be calculated.

$$= \frac{5 - n + 1}{1 + 2 + 3 + 4 + 5}$$

where n = rank for $chrom_i$

$$= \frac{6 - n}{15}$$

The cumulative probabilities listed in column 4 of Table 4.5 are used to select the chromosome. Table 4.6 shows the results for N_{keep} = 5 chromosomes.

Table 4.6

Rank Weighting

Number	**Chromosome**	P_n	$\sum_{i=1}^{n} P_i$
1	Chrom 1	0.333	0.333
2	Chrom 2	0.266	0.599
3	Chrom 3	0.2	0.799
4	Chrom 4	0.133	0.933
5	Chrom 5	0.066	0.999

In the beginning, we create a random number between zero and one using the observations in Table 4.5. The first chromosome, which has the largest cumulative probability of the random number, will enter the mating pool. If the random number is r = 0.477, then 0.333< r ≤ 0.599, and thus chromosome$_2$ is selected. Many alternatives can be followed for pairing with the same chromosome. First, let it go, implying that three of these chromosomes appear in the next generation. Second, we randomly pick another chromosome. The choice of this approach is more realistic with the problem under discussion, as randomness in this approach is natural.

b. **Cost weighting.** In this approach, rather than its rank in the population, the probability of selection is calculated from the cost of the chromosome. By subtracting the lowest cost of the discarded chromosomes $\left(C_{N_{keep+1}}\right)$ from the cost of all the chromosomes in the mating pool, a normalized cost is calculated for each chromosome.

$$C_n = c_n - c_{N_{keep+1}}$$

By the following formula, the probability of selection is calculated from the cost of the chromosome,

$$P_n = \left| \frac{C_n}{\sum_n^{N_{keep}} C_n} \right|$$

In this approach, a large spread in the cost between the top and bottom chromosomes will tend to weigh the top chromosome more. By contrast, the approach tends to weigh the chromosomes equally when all the chromosomes have approximately the same cost. Moreover, the probabilities must be recalculated for each generation when a chromosome is selected to mate with itself (Haupt & Haupt, 2004).

Universiti Utara Malaysia

4. **Tournament Selection.** In this approach, we randomly select a small subset of chromosomes (two or three) from the mating pool, and the chromosome with the lowest cost in this subset becomes a parent. Tournament selection works best in a large population because the population need not be sorted, and sorting becomes time consuming for large populations.

4.5.4 Mating

We define mating as the creation of one or more offspring from the selected parents in the pairing process. In this work, a combination of an extrapolation method with a 1X crossover method (single-point) was used. One or more crossover points in the selected chromosome chosen was considered as the simplest methods. Two parents were selected to produce two offspring. In two crossover points (2X) method, the variables between these points were swapped, that is, between parents. Crossover points were randomly selected, and then the variables in between were swapped.

In our problem, the first mating point was determined randomly by selecting a variable in the first pair of parents to be the crossover point:

$$Xp= \text{roundup}(\text{random}*N_{var})$$

$$Xp = \text{ceil}(\text{rand}(1,M)*3)$$

where

M = number of mating

For example, if M = 3, the we obtain three crossover points (Xp= 2, 3, 1).

Let

$$\text{Parent1}= [p_{m1}, p_{m2}, p_{m3},\ldots\ldots, p_{mxp},\ldots\ldots\ldots,p_{mNvar}]$$

$$\text{Parent2}= [p_{d1}, p_{d2}, p_{d3},\ldots\ldots\ldots, p_{dxp},\ldots\ldots\ldots,p_{dNvar}]$$

where m and d denote mom and dad.

Thereafter, a combination of selected variables was obtained to gain new variables as offspring.

Universiti Utara Malaysia

$$P_{new1} = p_{mxp} - \beta[p_{mxp} - p_{dxp}]$$

$$P_{new2} = p_{dxp} + \beta[p_{mxp} - p_{dxp}]$$

where β is also a random value between 0 and 1. The final step is to complete the crossover with the rest of the chromosomes.

$$offspring1 = [p_{m1}, p_{m2}, p_{m3}, \ldots\ldots, p_{new1}, \ldots\ldots\ldots, p_{dNvar}]$$

$$offspring2 = [p_{d1}, p_{d2}, p_{d3}, \ldots\ldots, p_{new2}, \ldots\ldots\ldots, p_{mNvar}]$$

If the first chromosome variable is selected, only the variables on the right of the selected variable are swapped. If the last variable of the chromosome is selected, only variables on the left of the selected variable are swapped. Offspring variables are not permitted outside the bounds set by the parent, unless $\beta > 1$.

If the first set of parents is given by

Chromosome1 = [0.1244, 0.4771],

Chromosome2 = [0.2346, 0.6783],

then a random number generator selects p_1 as the location of the crossover. The random number selected for β is 0.0373. The new offspring is given by:

Offspring1 = [0.1244 - 0.0373 * 0.1244 + 0.0373 * 0.2346, 0.6783]

=[0.0128 ,0.6783]

Offspring2 = [0.2346 + 0.0373 * 0.1244 - 0.0373 * 0.2346, 0.4771]

=[0.2304 ,0.4771]

4.5.5 Mutation

Random mutations can be described as an alternative to a particular percentage of the bits in a list of chromosomes. GA can explore mutation. It provides attributes that are not present in the original population and can prevent GA from converging too fast before sampling the entire cost surface. A single point mutation changes a 1 to a 0 in binary coding, and vice versa. Mutation points can be randomly selected from N_{pop}*N_{bits} total number of bits in a population matrix.

In the context of the Rocky Mountain National Park issue, 20% of the population (m=0.20) was selected for mutation (excluding the best chromosome). Therefore, seven pairs of random integers were selected using a random number generator, which correspond to the rows and columns of the mutated bits. The number of mutations is depicted using:

$$\#mutations = m * (N_{pop} - 1) * N_{bits} = 0.2 * 7 * 14 = 19.6 \cong 20$$

The following pairs were randomly selected:

mrow = [5 7 6 3 6 6 8 4 6 7 3 4 7 4 8 6 6 4 6 7]

mcol =[6 12 5 11 13 5 5 6 4 11 10 6 13 3 4 11 5 14 10 5]

The first random pair is (5, 6). Thus, the bit in row 5 and column 6 of the population matrix is mutated from a 1 to a ***0*** and so on.

In this work, we used the same concept to perform the mutation operator. The total number of mutations in the population matrix is multiplying the mutation rate (20%) by number of population, except for the best chromosome, which is multiplied by number of variable. For example, the number of mutations in population size 8 chromosome with two variables can be calculated using this formula.

#mutations = mutation rate*(Number of population-1) * Number of variables

#mutations = m_{rate} * (N_{pop} - 1) *N_{vars} = 0.2 * 7 *2 = 0.28 $\cong$ 3

Rows and columns of the variables to be mutated are selected by random numbers. Thus, a mutated variable is replaced by a new random variable.

Universiti Utara Malaysia

mrow = [54 7]

mcol = [32 1]

The first random pair is (5, 3). Thus, the value in row 5 and column 3 of the population matrix is replaced with a uniform random number between one and ten.

6.6130➔9.8190

In our problem, the random numbers in the population matrix has different ranges. The procedure below in (Figure 4.11) explains how to solve this point.

```
nmut=ceil((popsize-1)*Nt*mutrate)
for ii=1:nmut
ck=mcol(ii);
switchck
```

Figure 4.11. Mutation Procedure

4.6 Model Validation

Universiti Utara Malaysia

To examine the validity of the model, an appropriate statistical method such as cross validity is conducted. This refers to a model validation technique used in the assessment of findings of statistical analysis that can be generalized to an independent data set (Devijver & Kittler, 1982; Geisser, 1993; Kohavi, 1995). The method can be used in the estimation of the level of model fit to a set of data, which is independent of that data applied in order to train the model. The method is also employed in the estimation of any quantitative measure to fit that's suitable for the model data. Usually, it is used in contexts where the aim is forecasting and estimating how accurate a predictive model can perform. Cross-validation works by dividing a sample of data into complementary subsets and subsequently conducting an analysis on one of the subsets (the training set) and then validating the analysis against the other subset (validation or testing set). Several cross-validations can be conducted using various partitions to decrease the variability, and the validation findings are averaged over the rounds.

Mosteller and Doob (1949) says Cross-validation guards against hypotheses testing suggested by the data (named "Type III errors"), particularly when the samples are costly, hazardous, or impossible to gather.

Throughout this examination, cross-validation was performed using the K-fold technique and SPSS software. K-fold cross-examination involves a sample being divided at random into k equal sized sub-samples. Amongst the k sub-samples, one sub-sample is kept for the validation data during model testing whereas the other k – 1 samples become the training data (see Figure 4.12). The cross-validation process occurs k times and each of the k sub-samples is applied one time only as the validation data. The average of the k findings from the folds is averaged in order to provide estimation. The advantage of this technique compared to repeat random sub-sampling is that the observations can be applied for both training and validation. Also, every observation is applied for validation exactly one time.

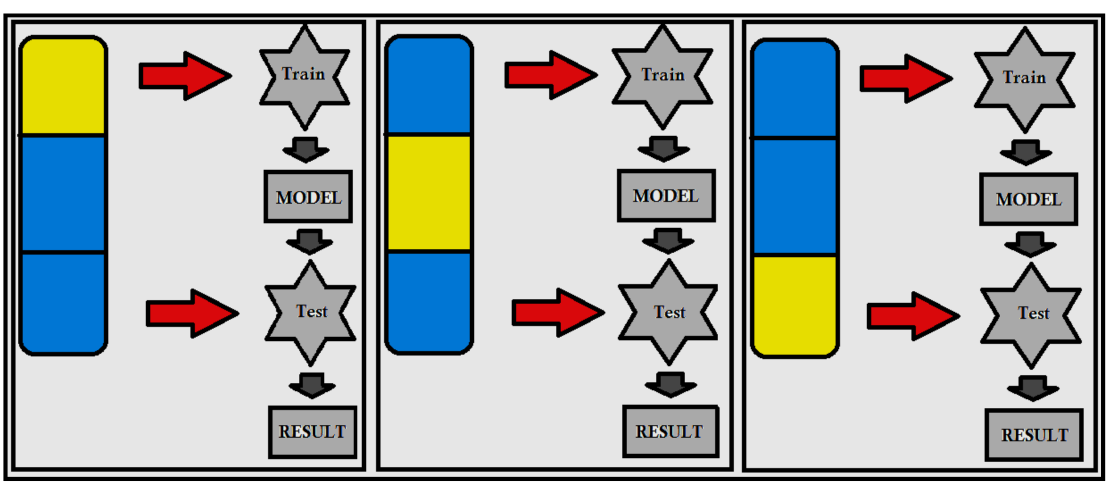


Figure 4.12. Procedure of Three-fold Cross-validation

Given that the sample size of this study was 63 and not large, the value of K is selected as 3 to provide adequate sample size in each partition. Moreover, CHAID method was used as growing method for this cross-validation. In CHAID analysis, the scales of predictors are banded into discrete groups prior to analysis.

Table 4.7 demonstrates the result of the cross validation, and shows the value of risk estimation for this study is 0.036. This risk estimation value indicated that knock predicted by the model was wrong only for 0.036% of the cases. Consequently, the risk of wrong prediction of knock for other situation was 0.036%. We can conclude the proposed model is applicable for approximately 99.063% of cases, implying that this model works properly for 99.063% of cases, and the prediction would be accurate.

Table 4.7

Estimation of Risk for Knock Prediction

Method	**Estimate**	**Std. Error**
Cross-Validation	0.036	0.005

Growing Method: CHAID

Dependent Variable: Knock

4.7 Summary and Discussion

In this chapter, the methodology described in the previous chapter was applied, which consists of three phases. Initially shows the behavior that has been the selection and data collection. In addition, clarifying how to build individual objectives and then building the evaluation function (model). While the last part shows the application of optimization methodology and verification of the effectiveness of the model.

After applied all these methodologies, final model (evaluation function) was obtained, as shown below.

Knock =-1*(0.24304*20.873*sin((29.873*tps-1405.39002)*pi/180)-1.08452*0.000117*rpm-0.000009226 * 38.29344 * sin((13.29344* temp+35.15755)*pi/180)+5)

It is easy to see that tradeoffs between these attributes are usually inevitable. Notably, when robustness and ease of use increase, efficiency typically decreases and vice versa. The stated general attributes are also interrelated. For instance, algorithms that are very sensitive to the selection of the tuning parameters are often problem dependent, and hence not sufficiently general. An algorithm tailored to a specific class of problems with common structure may be more efficient for that particular problem class than an algorithm that could be applied to a more general class of problems, but a lot less efficient or even unusable when applied to problems outside that class.

Universiti Utara Malaysia

CHAPTER FIVE
EXPERIMENTAL RESULTS AND ANALYSIS

5.1 Introduction

In this chapter will be divided into three parts: The first part shows the reasons analysis for choosing the three factors (Tps, Rpm,Temp), and discard another factors depending on the results that have been obtained through the application of some statistical measurements by (Minitab & SPSS) softwares, such that, was dealt the reasons for selecting the one of three models. The second part, using (Curve Expert Professional) program (CEP), to analysis the objectives in order to build the evaluation function, in addition to showing the results that have been obtained to validate the model chosen. Finally, Part three, by using (Matlab) program, shows the results of the work of the proposed model for the evaluation function in the optimization problem between conflicting objectives, and obtain a form, for simulations knocking problem in internal combustion engines.

5.2 Sample Size Testing

While known, factor analysis depend on the correlations structure between the factors, and it is know the correlation parameter value, depend on the sample size. Therefore it is important to test the sample size before doing the factors analysis.

Generally, judgment of the whether the sample size is enough or not, must use Kaiser-Meyer-Olkin(KMO) as a measure of sampling. The statistic value for this test is located in the range from zero to 1 integer; close this value to the 1 means increasing the reliability of factors that produced from analysis. Owner this test Kaiser (1974) refer to the minimum acceptable value for this statistic is 0.50 in order to judgment the sampling size enough or not. The result of this test (KMO) shown in Table 5.1 is 0.540 (greater than 0.50), this means our data enough to do this analysis (Williams, Brown, & Onsman, 2012).

Universiti Utara Malaysia

Table 5.1

KMO and Bartlett's Test

Kaiser-Meyer-Olkin Measure of Sampling Adequacy.		.540
Bartlett's Test of Sphericity	Approx. Chi-Square	.731
	df	3
	Sig.	.866

5.3 Selecting Factors and its Analysis, Results

By experimental of the study, three models was constructed by using general regression analysis, these models can help us to select the best factors that have effect on knocking.

- **Conditions of the method used to estimate the regression model parameters**

The method of least squares of the most famous methods in the estimation of the regression model parameters, the Conditions of this method is:

Universiti Utara Malaysia

1. Normality test: you can use F-test or T-test whether in test the overall significant or partial significant for regression model. Necessary provide the normality distribution for residuals.
2. Multi-collinearity: This means that there is a strong correlation and significance between two or more of the explanatory variables. Is one of the most important negative effects of the existence of multi-collinearity between explanatory variables is the instability of the regression coefficients.

5.3.1 General Regression Model Analysis: KNOCK versus TPS;TEMP;IGN;RPM

The construction of the first model took four factors that affect the phenomenon of knocking and characterized by the conflicting between them. Factors are Temperature (Temp), Revolution per minute (Rpm), Throttle position sensor (Tps), Ignition timing (IGN). After applying the general regression, and Analysis of Variance (ANOVA) see Table 5.2, were obtained the following results and regression equation is:

- **Regression Equation**

$$\text{KNOCK} = 988.455 - 12.3323\ \text{TPS} - 0.0164546\ \text{TEMP} + 0.00745664\ \text{IGN} + 0.000100153\ \text{RPM} \quad (5.1)$$

- **Analysis of Variance**

Table 5.2

Analysis of Variance (ANOVA)

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	1.234	4	.309	52.403	.000[b]
	Residual	.135	23	.006		
	Total	1.370	27			

a. Dependent Variable: KNOCK

b. Predictors: (Constant), RPM, TEMP, TPS, IGN

Through the following Table 5.3, note the P. value in factors TPS, RPM is the value of less than 0.05, and this reflects that there is a significant effect on knocking for these two factors. From other side, the rest factors (TEMP, IGN) P.value greater than 0.05 this reflects less effect on knocking.

Universiti Utara Malaysia

Table 5.3

Coefficients

No.	Term	Coef.	P. value
1	Constant	988.455	0.000
2	TPS	-12.332	0.000
3	TEMP	-0.0164	0.760
4	IGN	0.0074	0.122
5	RPM	0.0001	0.001

Therefore, to decrease the complex nature of the model (AIC simple defines the model complexity in terms of the number of free parameters) Bozdogan (2000), one must be deleted. In order to make the decision to delete any one of these factors, should be taking into account the correlation between all the factors, as well as the multicollinearity problem. Problems will also become apparent when predictor variables have strong correlation to each other. If this is the case, it is difficult to isolate the individual contribution of each predictor variable, which causes issues in the estimation of the relationship between the predictors and the outcome. Two statistics are used to diagnostics this problem (tolerance, Variance Inflation Factor (VIF)). For any variable in the model, if VIF is greater than 5, then this would be is a good pointer to such problem. The tolerance it is just reverse of VIF (Tolerance = 1/VIF) , this means , if tolerance is less than 0.2 , then this is a similarly a cause of concern(Field, 2009).

Table 5.4

Correlations TPS, IGN, TEMP, RPM

	TPS	**IGN**	**TEMP**
IGN	0.088		
TEMP	0.107	-0.033	
RPM	0.122	0.921	0.070

Through correlation analysis of the four factors was observed that there is a strong correlation between IGN and RPM where the 0.921, see Table 5.4 above, also the value of VIF for IGN and RPM (7.063, 7,126) respectively, see Table 5.5 below, more than 5, this would be a good reason to worry, so it is supposed to delete one of these factors with high correlation.

Table 5.5

Multi-collinearity problem for 4-factors

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	Collinearity Statistics	
		B	Std. Error	Beta			Tolerance	VIF
1	(Constant)	972.310	177.638		5.474	.000		
	TPS	-12.129	2.224	-.362	-5.454	.000	.974	1.026
	IGN	.007	.005	.276	1.582	.127	.142	7.063
	TEMP	-.017	.053	-.022	-.329	.745	.925	1.081
	RPM	.000	.000	.658	3.761	.001	.140	7.126

a. Dependent Variable: KNOCK

Universiti Utara Malaysia

The Summary of Model shown in the following Table 5.6.

Table 5.6

The Summary[b] of Model 1 for 4-Factors.

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Durbin-Watson
1	.949[a]	.901	.884	.076737	.351

a. Predictors: (Constant), RPM, TEMP, TPS, IGN

b. Dependent Variable: KNOCK

S = 0.0768620,   R-Sq = 90.08%,   R-Sq(adj) = 88.36% ,   PRESS = 0.204633
R-Sq(pred) = 85.07%

- **Durbin-Watson Statistic**

Durbin-Watson statistic = 0.345914

The normality test in Figure 5.1 Shown we find that the residuals are randomly distributed on both sides of the line, which means that the residuals are distributed moderate distribution (i e, follow normal distribution).

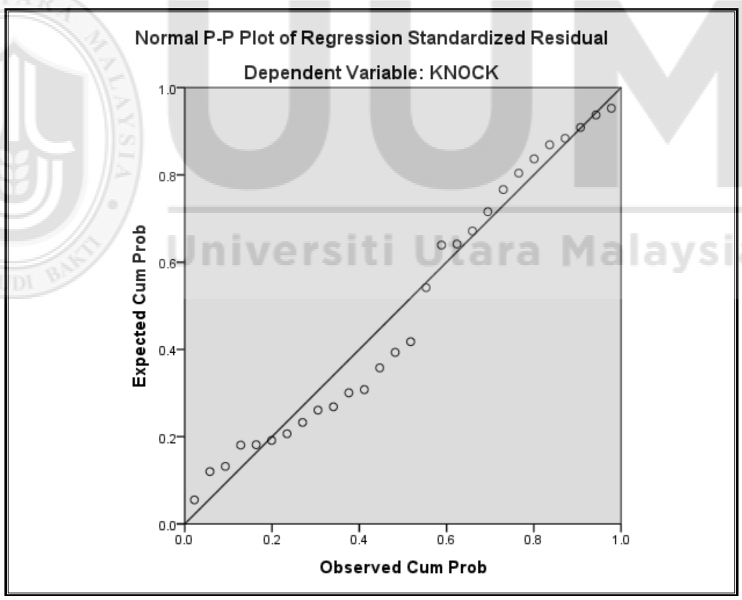


Figure 5.1. Normality Test for residual (Model 1)

5.3.2 General Regression Model Analysis: KNOCK versus TPS; TEMP; RPM

In the construction of the second model, were taken three factors have an effect on the phenomenon of knocking and characterized by the conflicting between them. Factors are Temperature (Temp), Revolution per minute (Rpm), Throttle position sensor (Tps). After applying the general regression, and Analysis of Variance (ANOVA), see Table 5.9, were obtained the following results and regression equation is:

- **Regression Equation**

KNOCK = 1001.73 - 12.4751 TPS - 0.0375599 TEMP + 0.000139913 RPM (5.2)

Also the value of VIF in table 5.7 for all factors less than 5.

Table 5.7

Multi-collinearity problem for 3-factors

Coefficients[a]

Model		Unstandardized Coefficients B	Unstandardized Coefficients Std. Error	Standardized Coefficients Beta	t	Sig.	Collinearity Statistics Tolerance	Collinearity Statistics VIF
2	(Constant)	986.580	182.881		5.395	.000		
	TPS	-12.285	2.290	-.367	-5.363	.000	.976	1.024
	TEMP	-.038	.053	-.049	-.721	.478	.986	1.015
	RPM	.000	.000	.915	13.411	.000	.982	1.018

a. Dependent Variable: KNOCK

Universiti Utara Malaysia

The Summary of Model shown in the following Table 5.8

Table 5.8

The Summary[b] *of Model 2 for 3-Factors*

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Durbin-Watson
2	.944[a]	.890	.877	.079104	.585

a. Predictors: (Constant), RPM, TEMP, TPS

b. Dependent Variable: KNOCK

S = 0.0793399 R-Sq = 88.97% R-Sq(adj) = 87.60%

PRESS = 0.217751 R-Sq(pred) = 84.11%

- **Analysis of Variance**

Table 5.9

Analysis of Variance (ANOVA)

ANOVA[a]

Model		Sum of Squares	df	Mean Square	F	Sig.
2	Regression	1.220	3	.407	64.967	.000[b]
	Residual	.150	24	.006		
	Total	1.370	27			

a. Dependent Variable: KNOCK

b. Predictors: (Constant), RPM, TEMP, TPS

- **Durbin-Watson Statistic**

Durbin-Watson statistic = 0.577961

The normality test in Figure 5.2 Shown we find that the residuals are randomly distributed on both sides of the line, which means that the residuals are distributed moderate distribution (i e, follow normal distribution).

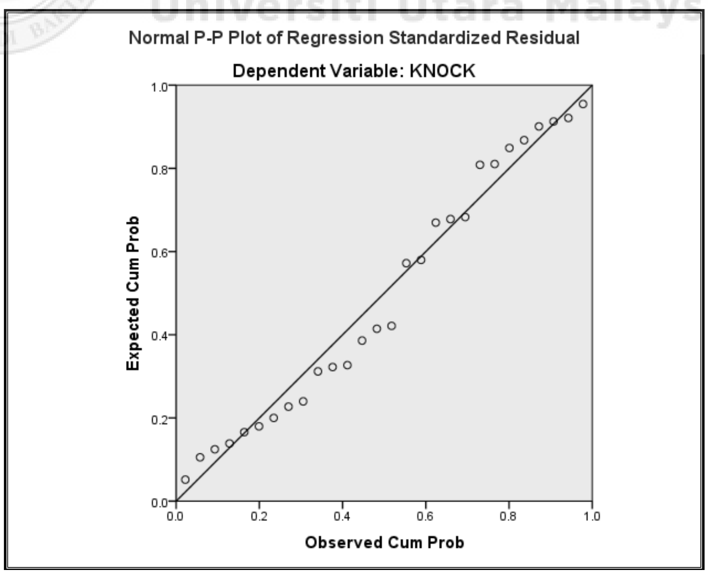


Figure 5.2. Normality Test for residual (model 2)

5.3.3 General Regression Model Analysis: KNOCK versus TPS; TEMP; IGN

In the construction of the third model, were taken three factors have an effect on the phenomenon of knocking and characterized by the conflicting between them. Factors are Temperature (Temp), Throttle position sensor (Tps), Ignition timing (IGN). After applying the general regression, and Analysis of Variance (ANOVA), were obtained the following results and regression equation is:

- **Regression Equation**

$$KNOCK = 929.707 - 11.6524\ TPS + 0.0333036\ TEMP + 0.0235528\ IGN \qquad (5.3)$$

Also the value of VIF in table 5.10 for all factors less than 5.

Universiti Utara Malaysia

Table 5.10

Multi-collinearity problem for 3-factors

Coefficients[a]

Model		Unstandardized Coefficients B	Unstandardized Coefficients Std. Error	Standardized Coefficients Beta	t	Sig.	Collinearity Statistics Tolerance	Collinearity Statistics VIF
3	(Constant)	911.808	220.081		4.143	.000		
	TPS	-11.428	2.757	-.341	-4.145	.000	.981	1.019
	TEMP	.032	.064	.042	.509	.615	.987	1.013
	IGN	.024	.002	.882	10.769	.000	.991	1.009

a. Dependent Variable: KNOCK

The Summary of Model shown in the following Table 5.11

Table 5.11

Summary[b] of Model 3 for 3-Factors

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Durbin-Watson
3	.917[a]	.840	.820	.095464	.619

a. Predictors: (Constant), IGN, TEMP, TPS

b. Dependent Variable: KNOCK

S = 0.0954181   R-Sq = 84.05%   R-Sq(adj) = 82.06%

PRESS = 0.286746 R-Sq(pred) = 79.07%

- **Analysis of Variance**

Table 5.12

Analysis of Variance (ANOVA[a])

Model		Sum of Squares	df	Mean Square	F	Sig.
3	Regression	1.151	3	.384	42.101	.000[b]
	Residual	.219	24	.009		
	Total	1.370	27			

a. Dependent Variable: KNOCK

b. Predictors: (Constant), IGN, TEMP, TPS

- **Durbin-Watson Statistic**

Durbin-Watson statistic = 0.623352

The normality test in Figure 5.3 Shown we find that the residuals are randomly distributed on both sides of the line, which means that the residuals are distributed moderate distribution (i e, follow normal distribution).

Universiti Utara Malaysia

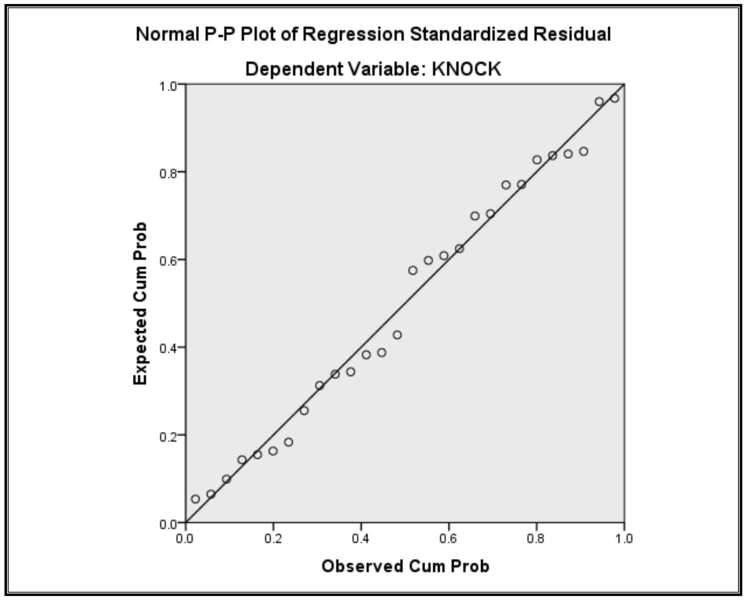


Figure 5.3. Normality Test for residual (Model 3)

After obtaining the previous results for the three models, can be summarized in the following Table 5.13, in order to select the best model, among those three models.

Table 5.13

Summary result for three Models

Model	**Adj. R^2**	**Std. error of the estimation**	**F-Test**	**Sig. P-value**	**Durbin Watson**
Model 1	0.884	0.0767	52.403	0.000	0.351
Model 2	0.877	0.0790	64.967	0.000	0.585
Model 3	0.820	0.0954	42.101	0.000	0.619

Three models are constructed by using regression analysis:

$$Knock = 988.455 - 12.3323\ TPS - 0.0164546\ TEMP + 0.00745664\ IGN + 0.000100153\ RPM \quad (5.1)$$

$$Knock = 1001.73 - 12.4751\ TPS - 0.0375599\ TEMP + 0.000139913\ RPM \quad (5.2)$$

$$Knock = 929.707 - 11.6524\ TPS + 0.0333036\ TEMP + 0.0235528\ IGN \quad (5.3)$$

Through the above Table 5.8 , we note a range of situations have been studied and analyzed in order to reach the best model through the selection of the factors that make the model less complexity. Note the P. value in all cases is the value of less than 0.05, and this reflects that there is a significant effect on knocking for all cases.

Choose the best model among all 3 models, we used appropriate statistical criteria Akaike Information Criterion (AIC). AIC is the measurement of relative quality of a model for a set of data that can provide a tool for the selection of a model. AIC handles the trade-off between the complexity and goodness of fit. AIC can be calculated using the following formula:

$$AIC=2k-2\ln(L)$$

Where, K is the number of factors, L is the maximized value of the likelihood function of the estimated model.

In other form:

$$AIC= n * \ln(\partial^2) + 2 * (K+1)$$

where, n is size of sample, K is the number of factors (predictors) in given statistical model and (∂^2 = SSE/ n).

Table 5.14

AIC Computation Results for Three Models

Model	n(Sample Size)	SSE	∂(segma)^2	K	AIC-Value
TPS,TEMP,RPM,IGN	28	0.135	0.004821429	4	-139.3711803
TPS,TEMP,RPM	28	0.15	0.005357143	3	-138.4210859
TPS,TEMP,IGN	28	0.219	0.007821429	3	-127.8248657

From the result above in Table 5.14, since, the complexity defines in AIC is the number of free parameters, as well as AIC score difference between two models is in magnitude of 1-2, the difference is significant, therefore, we can omit the first model (Bozdogan, 2000).

According to the relative probability that the i^{th} model minimizes the (estimated) information loss, we can choose it. If we wrote the relative likelihood of the model to reduce the loss of information as follows:

$A=e^b$

$b = ((AIC_{min}-AIC_i)/2)$

We can calculate the value of A as we vary b.

$A= Exp((AIC_{min} – AIC_i)/2)$,

From the table 5.14 can take the three AIC-value to calculate A and get the result of decision as shown:

We would omit the third model from further consideration. Then the second model is:

$\exp((-139.3711 - (-138.4210)/2)) = \exp(-0.4755)$

Such that , e= 2.718 “natural exponential”, $A = 2.718^{-0.4755}$

$A=e^b=0.6219$

That means, 0.6219 times as probable as the first model to minimize the information loss. Similarly, the third model is exp((-139.3711 − (-127.824))/2) = exp(-57735) = 0.0031 times as probable as the first model to minimize the information loss.

Initially delete IGN factor from the proposed model after that, the remaining three factors namely RPM, TPS and TEMP are analyzed. From table 5.13, where it was noted that the explanatory power of the model (Adjusted R Square) has decreased to 0.877 after it was 0.884, a value slightly. We also note that the value of the F-test has risen from 52.40 to 64.96 and this is proof model improved by good if compared with the value before pulling factor IGN. Note that in table 5.8 the value of the estimate error has increased from 0.07674 to 0.07910 a slight amount.

Secondly, taking into consideration the second case, which deleted the RPM factor of the model as the study and analysis of the situation has been shown that the explanatory power of the model has decreased from 0.884 to 0.820.We note also that the value of F-test has fallen from 52.403 to 42.101, a relatively large value, while increasing the standard error of the model from 0.07674 to 0.09546. Through analysis of the results it is clear that the second model is better than the third model, which means, deleted factor IGN.

5.4 Results and analysis of constructing objective functions

After the decision to identify the factors that will be involved in the construction of the evaluation function by the results which obtained from the analysis of the factors tested, was construct individual objectives for each factor.

5.4.1 TPS Objective Results

Depending on the data has been the objective building. Figure 5.4 below, represents the scatter of the raw data, which represents the objective TPS.

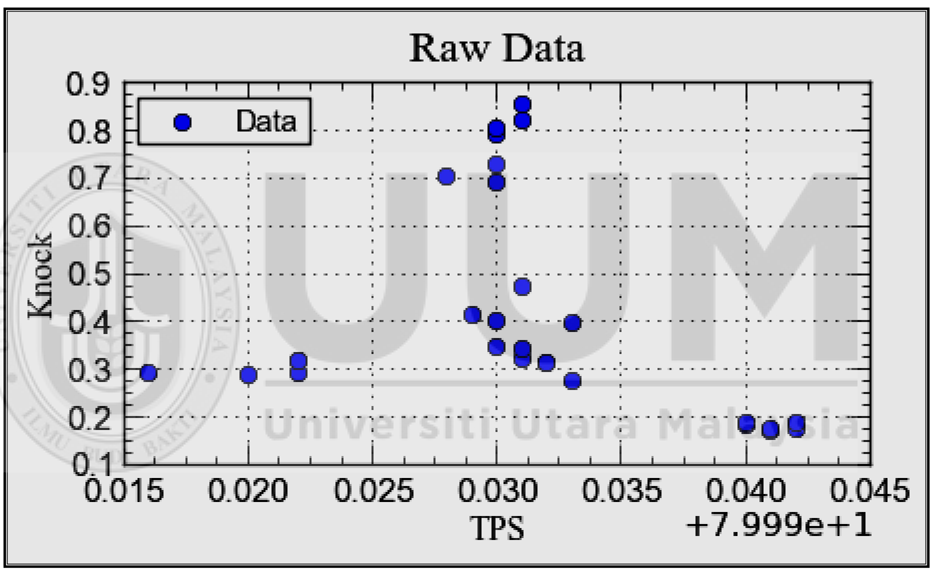


Figure 5.4. Scatter Raw Data for TPS objective

After the applying of nearly 50 models was obtained four best candidate models for simulations of such data, see Table 5.15

Table 5.15

Some models are applied on raw data

Name	Kind	Family	Score	R	R^2	Std_Err	AICC
Sinusoidal	Regression	Miscellaneous	525	0.733239	0.537640	0.162446	-99.091172
Gaussian Model	Regression	Miscellaneous	489	0.690112	0.476255	0.169400	-98.120672
Heat Capacity	Regression	Miscellaneous	461	0.653144	0.426597	0.177249	-95.584353
Steinhart-Hart Equation	Regression	Miscellaneous	438	0.619208	0.383419	0.183802	-93.551510
Reciprocal Quadratic	Regression	Yield-Density Models	438	0.619214	0.383426	0.183801	-93.551814
Logistic Power	Regression	Sigmoidal Models	401	0.548592	0.300954	0.195707	-90.036744
Ratkowsky Model	Regression	Sigmoidal Models	401	0.547955	0.300255	0.195805	-90.008775
Rational Model	Regression	Miscellaneous	377	0.494487	0.244517	0.207649	-85.342832
Rational Model	Regression	Miscellaneous	377	0.494487	0.244517	0.207649	-85.342832
Natural Logarithm	Regression	Exponential Models	339	0.262106	0.068700	0.221504	-84.330678
Wavy	Regression	Custom	337	0.233177	0.054372	0.223202	-83.903184
Normal (Gaussian) PDF	Regression	Distribution Models	337	0.228152	0.052053	0.223475	-83.834623
Saturation Growth Rate	Regression	Growth Models	335	0.192009	0.036867	0.225258	-83.389622
Reciprocal	Regression	Yield-Density Models	335	0.192059	0.036887	0.225256	-83.390180
Piecewise Linear	Regression	Custom	333	0.262139	0.068717	0.230547	-79.485040
Bleasdale	Regression	Yield-Density Models	333	0.219354	0.048116	0.228374	-81.392412
Logistic	Regression	Sigmoidal Models	332	0.192009	0.036867	0.229719	-81.063467

The top result can be summarize in Figure 5.5

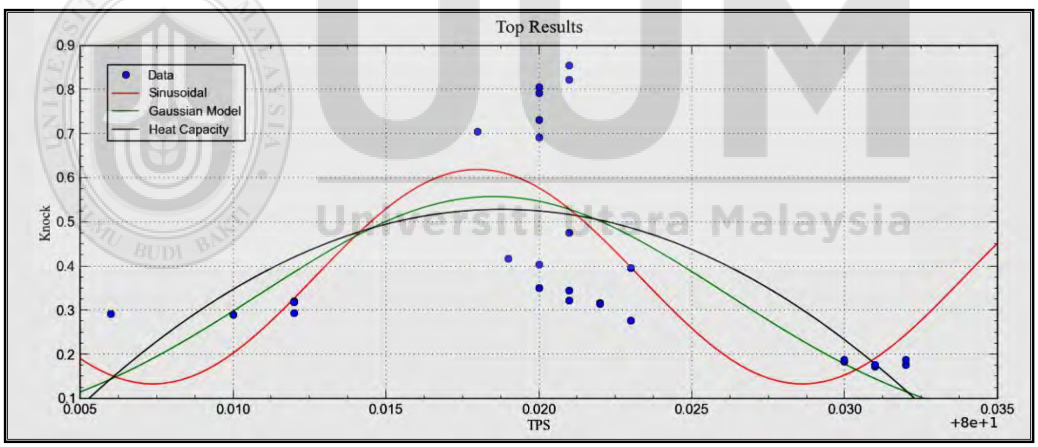


Figure 5.5. 3-Top Results Models for TPS objective

Note from the results, in Table 5.15, that best fit between the tested models and our data, the model (sinusoidal), see Figure 5.6 is the best model simulates factor data (TPS) as the value(-99.09) of (AIC) less than the rest of the other models, In addition to the lowest standard error. While the explanatory power ($r^{\wedge 2}$) is the highest, compared with other models.

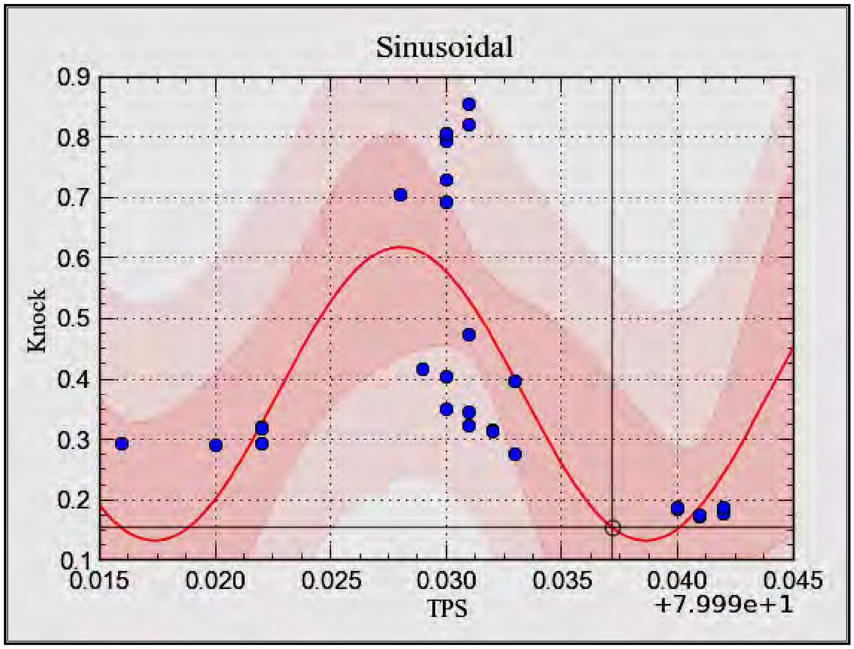


Figure 5.6. Best fit model for TPS raw data

Universiti Utara Malaysia

We also note from the Table 5.15, there is a great close between the model (Sinusoidal) and (Gaussian model) in terms of the different values such as AIC, standard error and the explanatory power ($r^{\wedge 2}$), this differentiation, will discuss in next chapter as well as comparing between them.

An appropriate model is selected and tested, then a Wald-Wolfowitz test is run on the residuals. Figure 5.7 outlines the observed number of runs (nruns = 12), as well as the likelihood (22.63%) that the observed quantity of runs may occur if the model used to fit the data is correct (i.e., the residuals are randomly distributed around the curve). The run pattern of the residuals is unlikely if the probability is less than 5%. The pattern is not unlikely if it’s greater than 5% (which is different to being likely, which cannot be claimed). A higher likelihood is more desirable. The residual plot also shows a straight-line fit to the residual points. The light red regression line depicts whether the trend is upwards or downwards (depending on the slope) and also shows if the residuals bias upward or downward.

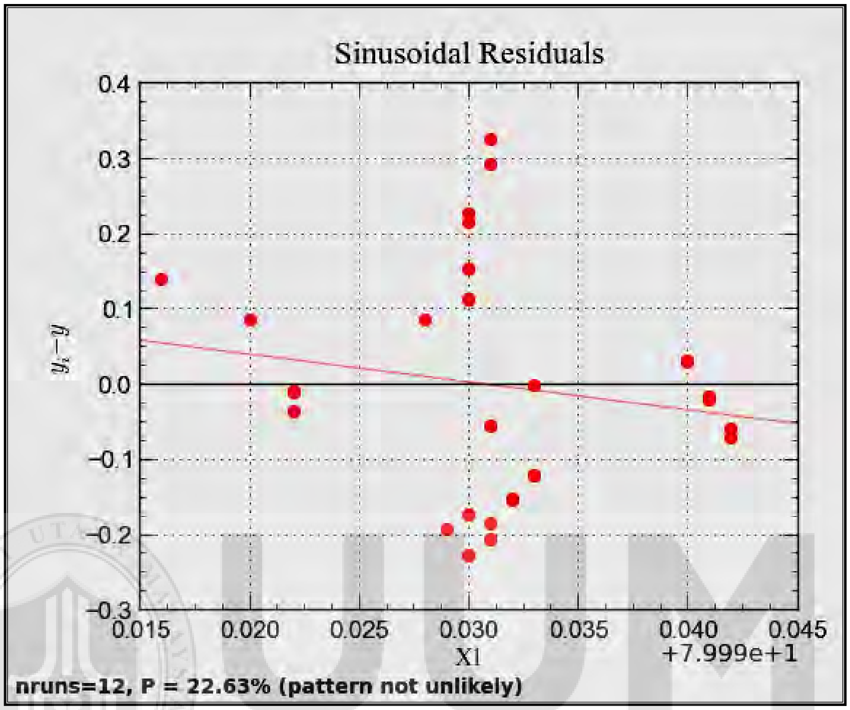


Figure 5.7. Test Residual randomness for TPS fitting model

Universiti Utara Malaysia

A graph depicting the convergence history of non-linear regressions is displayed. The difference between results and data (known as the norm) is depicted as a function of iteration number, and the change in the residual is also shown as a function of iteration number (Figure 5.8). The residual of the last iteration should be to the level set in the application preferences if the iteration converges, except in the case of the iteration terminating due to a lack of change in the parameters, instead of a lack of change in the residual.

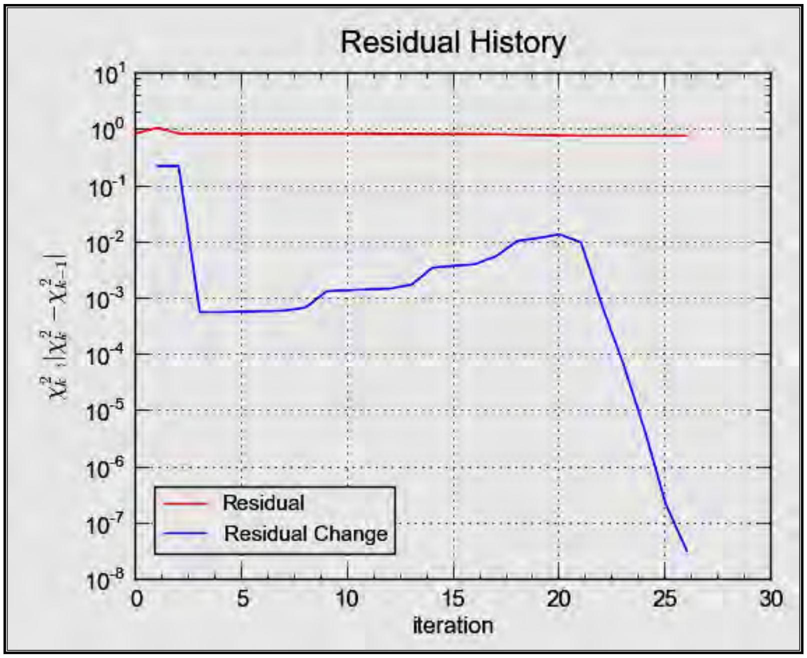


*Figure 5.8.*Convergence History for TPS factor

A graph depicting the parameter history of non-linear regressions is displayed. Each parameter value is depicted as a function of iteration number. It is evident whether the parameters have 'settled' on a certain value before termination of the iteration. The parameters remain flat to the right side of the plot, which indicates they have settled, unless the iteration terminates because it exceeds the maximum number of iterations (Figure 5.9).

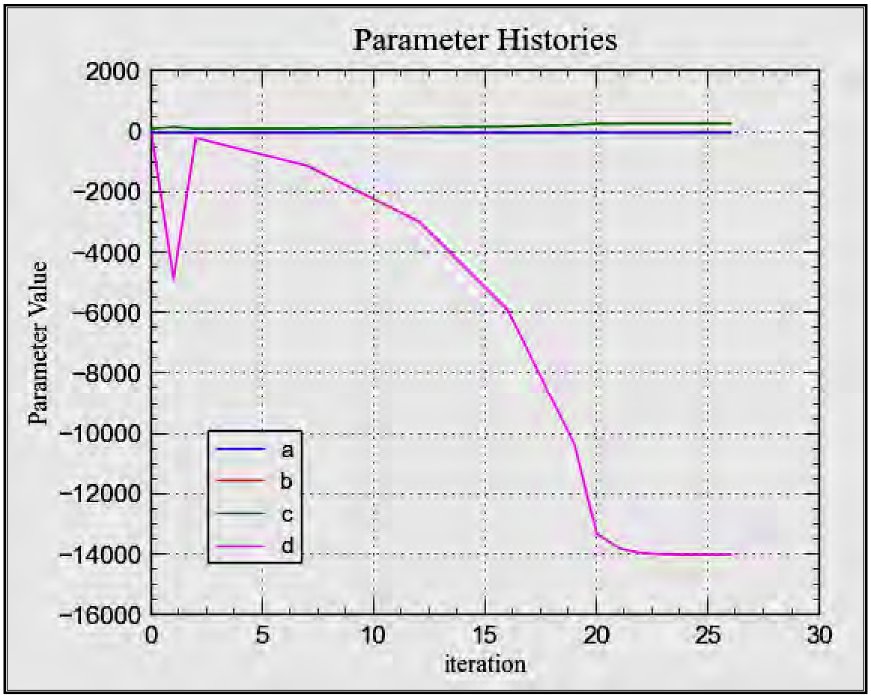


Figure 5.9. Parameter histories for TPS factor

Figure 5.10 depicts a confidence band, which is the area where a certain likelihood (usually 95%, but can be adjusted) of containing the true curve which fits the data. The prediction band is an area that possesses likelihood (typically 95%, can be adjusted) of containing future data point. The prediction band is wider than the confidence band.

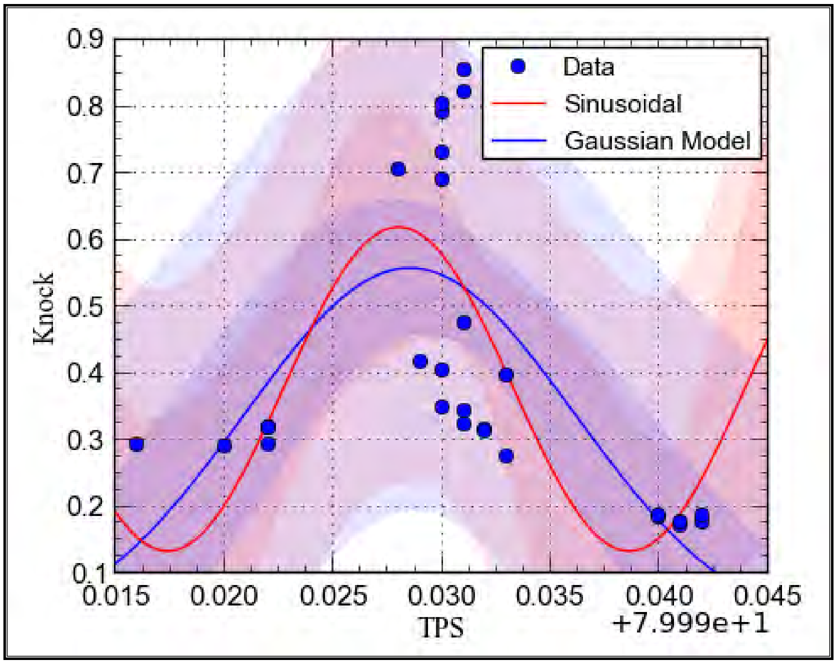


Figure 5.10. Confidence band and Prediction band

UUM
Universiti Utara Malaysia

Finally, the overview for TPS objective is :

Overview

Name	Sinusoidal
Kind	Regression
Family	Miscellaneous
Equation	y = a + b*cos(c*x + d)
# of Indep. Vars	1
Standard Error	0.162446
Correlation Coeff. (r)	0.733239
Coeff. of Determination (r^2)	0.537640
DOF	24
AICC	-99.091172

Parameters

Name	Value
a	3.765521870716789E-01
b	2.430463750916539E-01
c	2.958772810209356E+02
d	-1.400568045830553E+04

5.4.2 RPM Objective Results

The same procedure was applied to the rest of the factors data (RPM, TEMP.) were obtained results shown below. In Figure 5.11, represents the scatter raw data for RPM objective.

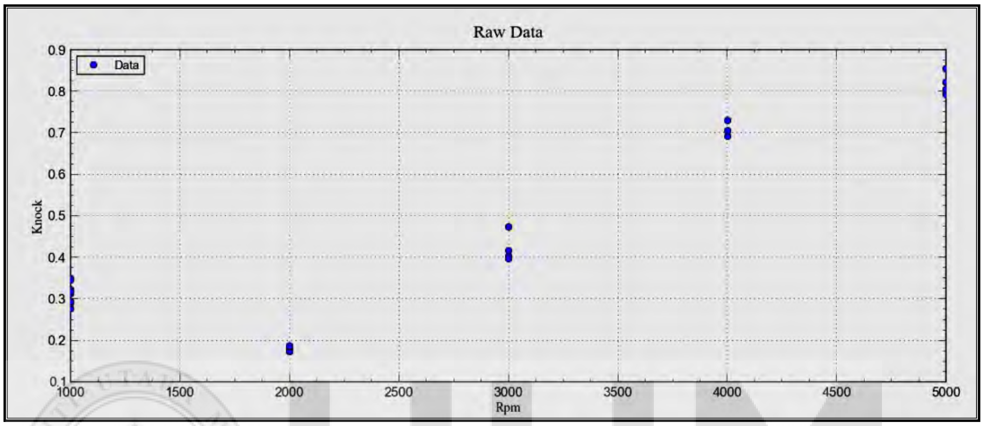


*Figure 5.11.*Scatter Raw Data for RPM objective

Through the results shown in Table 5.16 and overview below, we note there a rapprochement between the two models (Polynomial regression degree 3) and (Sinusoidal). Results also showed after a comparison between the two models, the first model (Polynomial Deg.3) is the best of the second model (Sinusoidal) for several reasons will be discussed in the next chapter.

Overview

	Sum of Squares	DOF	AICC
Polynomial Regression (degree=3)	0.0127826	24	-208.373
Sinusoidal	0.0150047	24	-203.885

UUM Universiti Utara Malaysia

Table 5.16

Some models are applied on RPM raw data

Results

Name	Kind	Family	Score	R	R^2	Std_Err	AICC
Polynomial Regression (degree=3)	Regression	Linear Regressions	973	0.995323	0.990668	0.023078	-208.372547
Sinusoidal	Regression	Miscellaneous	970	0.994508	0.989046	0.025004	-203.884717
Heat Capacity	Regression	Miscellaneous	931	0.981481	0.963305	0.044839	-172.555198
DR-Hill	Regression	Dose-Response Models	894	0.967843	0.936720	0.060097	-154.776992
Polynomial Regression (degree=2)	Regression	Linear Regressions	835	0.941853	0.887087	0.078655	-141.083556
Weibull Model	Regression	Sigmoidal Models	823	0.936630	0.877277	0.083692	-136.230678
Steinhart-Hart Equation	Regression	Miscellaneous	786	0.917590	0.841972	0.093051	-131.671200
Reciprocal Quadratic	Regression	Yield-Density Models	785	0.917036	0.840955	0.093350	-131.491504
Hyperbolic Decline	Regression	Decline Models	781	0.915153	0.837505	0.094357	-130.890767
Bleasdale	Regression	Yield-Density Models	781	0.915153	0.837505	0.094357	-130.890767
Exponential Decline	Regression	Decline Models	775	0.911457	0.830753	0.094427	-132.076929
Exponential	Regression	Exponential Models	775	0.911457	0.830753	0.094427	-132.076929
Ratkowsky Model	Regression	Sigmoidal Models	774	0.911421	0.830689	0.096316	-129.740150
Richards	Regression	Sigmoidal Models	773	0.911533	0.830892	0.098243	-127.253738
Farazdaghi-Harris	Regression	Yield-Density Models	773	0.910916	0.829767	0.096577	-129.588188
Harmonic Decline	Regression	Decline Models	772	0.909930	0.827973	0.095200	-131.620814
Reciprocal	Regression	Yield-Density Models	772	0.909930	0.827973	0.095200	-131.620814
Rational Model	Regression	Miscellaneous	772	0.911080	0.830066	0.098482	-127.117403
Reciprocal Logarithm	Regression	Exponential Models	753	0.899390	0.808902	0.100338	-128.676897
Linear	Regression	Linear Regressions	698	0.867065	0.751802	0.114350	-121.356589
Exponential Association 3	Regression	Growth Models	696	0.866683	0.751140	0.116770	-118.955892
Piecewise Linear	Regression	Custom	695	0.867065	0.751802	0.119019	-116.510435
DR-Hill-Zerobackground	Regression	Dose-Response Models	665	0.846758	0.717000	0.124522	-115.356274
Logistic Power	Regression	Sigmoidal Models	665	0.846758	0.717000	0.124522	-115.356280
MMF	Regression	Sigmoidal Models	655	0.841236	0.707678	0.129166	-111.928895

Universiti Utara Malaysia

Fitting raw data results, with the polynomial regression model showed in Figure 5.12 and the following overview:

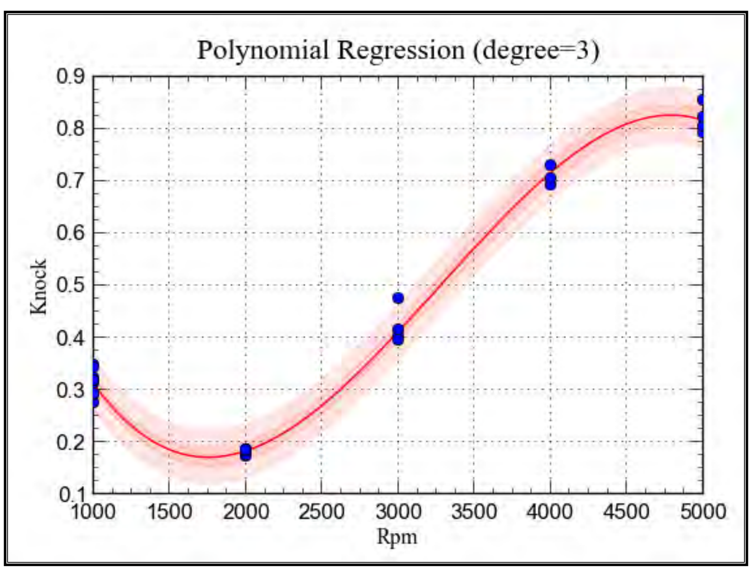


*Figure 5.12.*Polynomial Regression Model fitted with RPM data

Overview

Name	Polynomial Regression (degree=3)
Kind	Regression
Family	Linear Regressions
Equation	y = a + b*x + c*x^2 + ...
# of Indep. Vars	1
Standard Error	0.023078
Correlation Coeff. (r)	0.995323
Coeff. of Determination (r^2)	0.990668
DOF	24
AICC	-208.372547

Parameters

Name	Value
a	1.074740235484088E+00
b	-1.174947740852971E-03
c	4.582490247071722E-07
d	-4.671897118323384E-11

The Wald-Wolfowitz test is performed on the residuals. The results, observed number of runs (nruns=12) and the likelihood (16.78%) that this observed number of runs could occur if the model was used to fit the data was correct, is at Figure 5.13. The run pattern of the residuals is unlikely if the probability is less than 5%. The pattern is not unlikely if the probability is greater than 5% (which is different to being likely, which is not claimed). A higher likelihood is desirable. A straight line fit to the residual points is also displayed on the residual plot. The light red regression line shows whether the trend is upwards or downwards to the data, determined by the slope of the line, as well as depict whether the residuals bias upward or downward.

Universiti Utara Malaysia

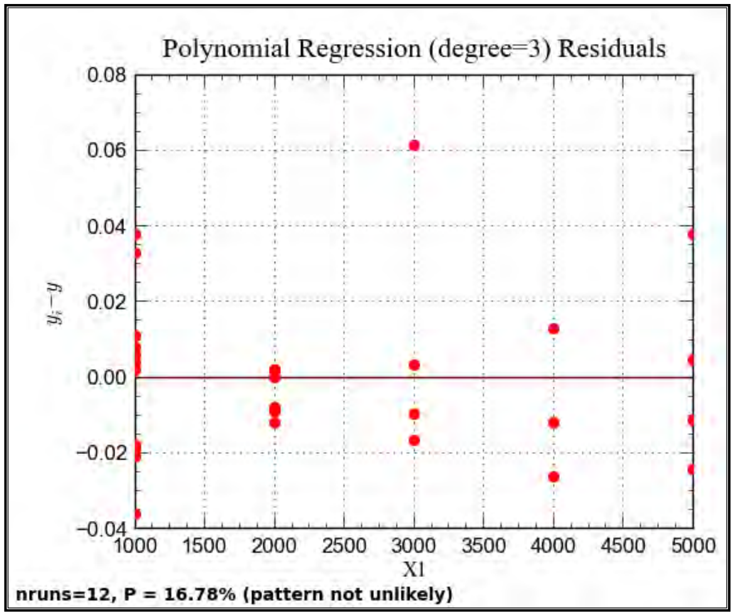


*Figure 5.13.*Test Residual randomness for RPM fitting model

5.4.3 Temperature Objective Results

Depending on the data has been the objective building. Figure 5.14 below, represents the scatter of the raw data, which represents the objective TEMP.

Universiti Utara Malaysia

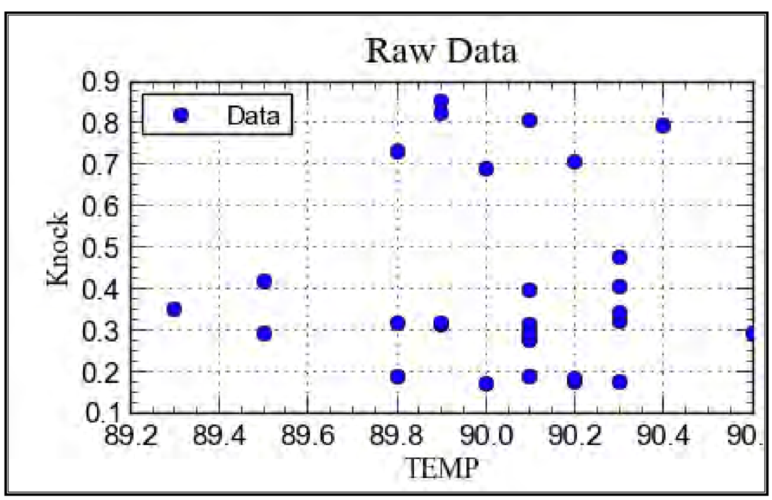


*Figure 5.14.*Scatter Raw Data for TEMP objective

After the applying of nearly 50 models was obtained four best candidate models for simulations of such data, see Table 5.17

Table 5.17

Some models are applied on TEMP raw data

Results

Name	Kind	Family	Score	R	R^2	Std_Err	AICC
Sinusoidal	Regression	Miscellaneous	337	0.304206	0.092541	0.227579	-80.210671
Log Normal PDF	Regression	Distribution Models	333	0.128723	0.016570	0.227619	-82.805659
Normal (Gaussian) PDF	Regression	Distribution Models	333	0.128375	0.016480	0.227630	-82.803113
Wavy	Regression	Custom	332	0.070429	0.004960	0.228959	-82.477059
Lowess Smoothing	Smoother	Smoothing	332	0.194789	0.037943	0.229591	-81.094750
Lowess Smoothing	Smoother	Smoothing	332	0.194789	0.037943	0.229591	-81.094750
Exponential Decline	Regression	Decline Models	331	0.019796	0.000392	0.229484	-82.348801
Harmonic Decline	Regression	Decline Models	331	0.014583	0.000213	0.229505	-82.343781
Log Normal CDF	Regression	Distribution Models	331	0.000000	0.000000	0.229529	-82.337825
Normal (Gaussian) CDF	Regression	Distribution Models	331	0.000000	0.000000	0.229529	-82.337825
Modified Power	Regression	Power Law Family	331	0.022616	0.000511	0.229470	-82.352151
Exponential	Regression	Exponential Models	331	0.022616	0.000511	0.229470	-82.352151
Exponential Association 2	Regression	Growth Models	331	0.000000	0.000000	0.229529	-82.337826
Reciprocal	Regression	Yield-Density Models	331	0.021717	0.000472	0.229475	-82.351035
Natural Logarithm	Regression	Exponential Models	331	0.023355	0.000545	0.229466	-82.353103
Modified Exponential	Regression	Exponential Models	331	0.021206	0.000450	0.229477	-82.350421
Saturation Growth Rate	Regression	Growth Models	331	0.021210	0.000450	0.229477	-82.350425
Geometric	Regression	Power Law Family	331	0.022664	0.000514	0.229470	-82.352212
Modified Geometric	Regression	Power Law Family	331	0.017865	0.000319	0.229492	-82.346764
Power	Regression	Power Law Family	331	0.000000	0.000000	0.229543	-82.334450
Harmonic Decline	Regression	Imported	331	0.014583	0.000213	0.229505	-82.343781
Linear	Regression	Linear Regressions	331	0.023625	0.000558	0.229465	-82.353459
Reciprocal Quadratic	Regression	Yield-Density Models	330	0.131855	0.017386	0.232031	-80.502751
Heat Capacity	Regression	Miscellaneous	329	0.125633	0.015784	0.232220	-80.457139
Gaussian Model	Regression	Miscellaneous	329	0.128644	0.016549	0.232130	-80.478931

Universiti Utara Malaysia

The top result can be summarize in Figure 5.15

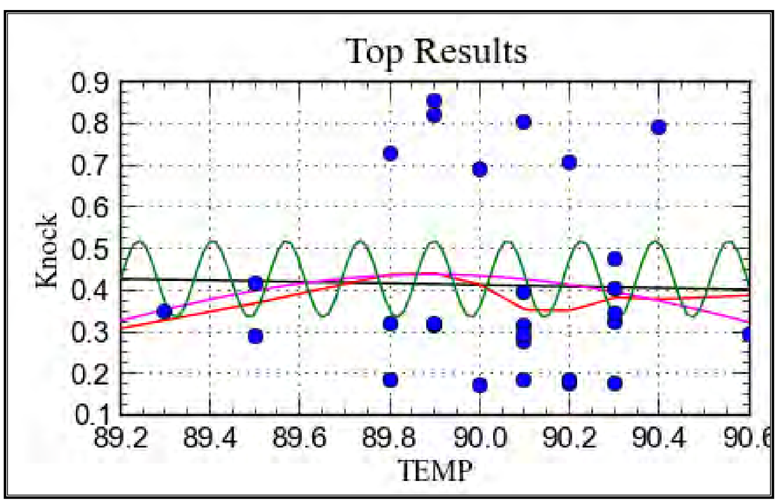


Figure 5.15. 4-Top Results Models for TEMP objective

Note from the results, in Table 5.15, that best fit between the tested models and our data, the model (sinusoidal), see Figure 5.16 is the best model simulates factor data (TEMP) as the value(-80.2106) of (AIC) less than the rest of the other models, In addition to the lowest standard error. While the explanatory power ($r^{\wedge 2}$) is the highest, compared with other models.

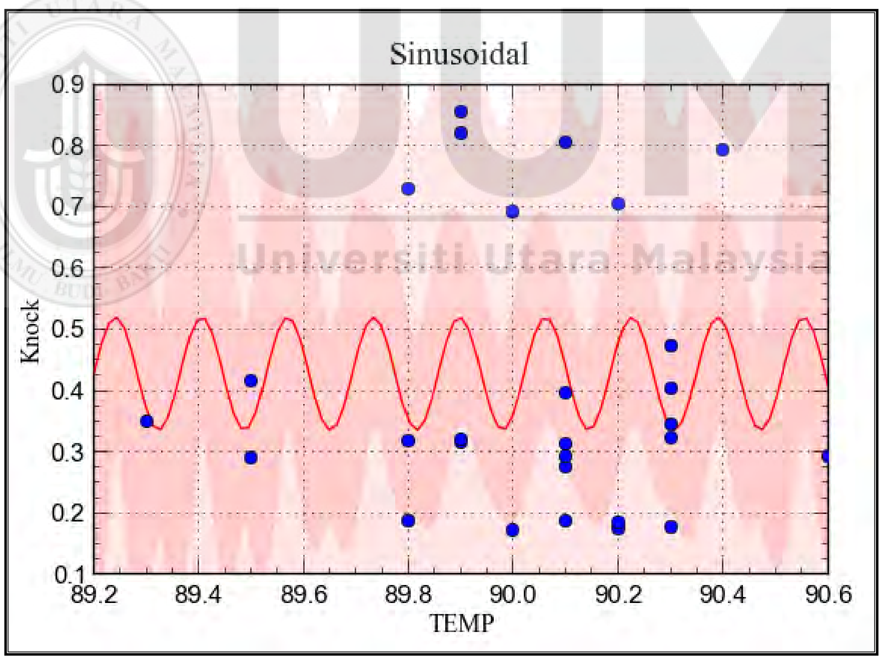


Figure 5.16. Sinusoidal Regression Model fitted with RPM data

Universiti Utara Malaysia

Parameters

Name	Value
a	4.285813243847063E-01
b	9.226621719132643E-02
c	3.829344361692780E+01
d	3.215755032763011E+01

Overview

Name	Sinusoidal
Kind	Regression
Family	Miscellaneous
Equation	y = a + b*cos(c*x + d)
# of Indep. Vars	1
Standard Error	0.227579
Correlation Coeff. (r)	0.304206
Coeff. of Determination (r^2)	0.092541
DOF	24
AICC	-80.210671

The Wald-Wolfowitz runs test is conducted on the residuals. The result, observed number of runs (nruns = 17), and likelihood (80.69%) are depicted at the bottom of Figure 5.17. A higher likelihood is preferable. The result displayed on the residual plot is a straight-line fit to the residual points. The light red regression line depicts a downwards trend and a residual bias downwards.

UUM Universiti Utara Malaysia

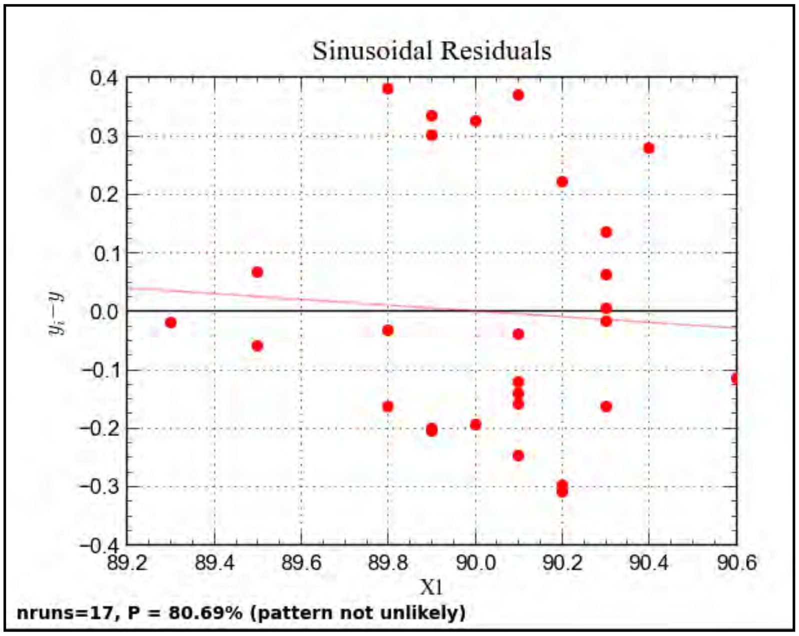


*Figure 5.17.*Test Residual randomness for TEMP fitting model

The results showed that the residual (error) resulting from the fitting process between the model and the data is (0.2275), and in the following Figure 5.18 below shows the change in the amount of error.

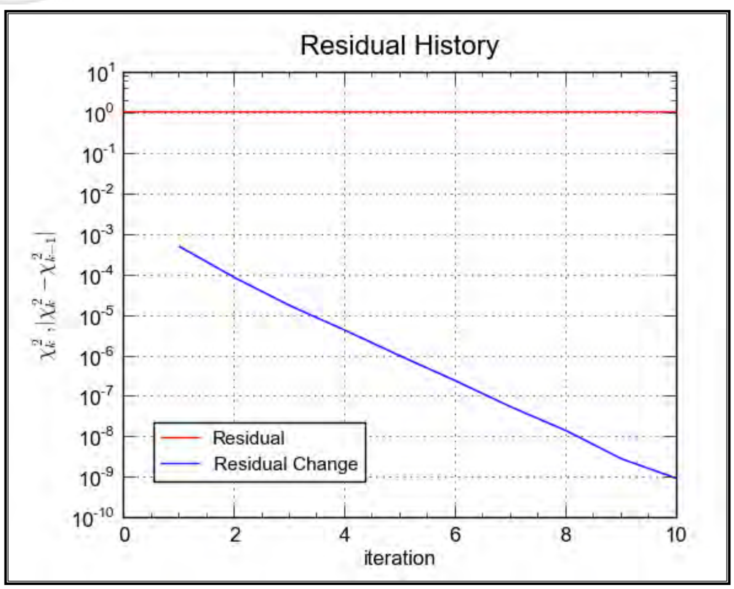


*Figure 5.18.*Cconvergence History for TEMP factor

For nonlinear regressions, the parameter history is graphically displayed. Each parameter value is depicted as a function of iteration number. It is possible to determine if the parameters ‘settled’ on a particular value prior to iteration termination. The parameters are always flat at the right side of the plot, unless the iteration terminated because it exceeded the maximum number of iterations set in the application preferences. This indicates that they are settled (as per Figure 5.19).

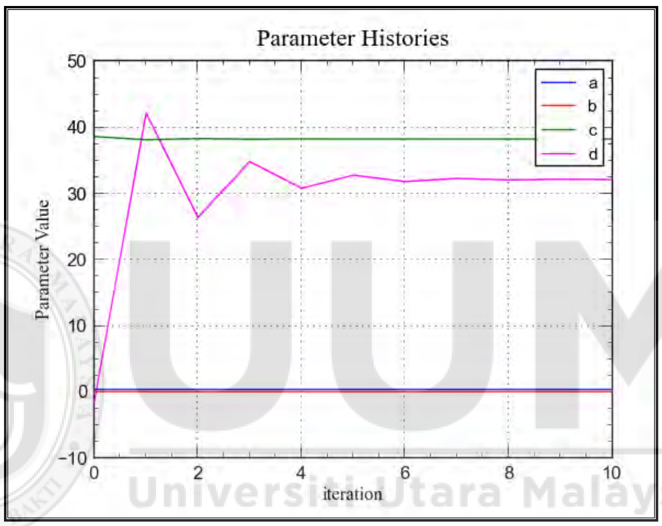


*Figure 5.19.*Parameter histories for TEMP factor

Generally, can summarize the results above in Table 5.18 for three factors as a individual objectives.

Table 5.18

Summary Three best models

M	Name	Kind	Family	Equation
1	Sinusoidal	Regression	Miscellaneous	Y=a+b*cos(c*TPS+d)
2	Polynomial Deg.3	Regression	Linear Regression	$Y=a+b*RPM+c*RPM^2+d*RPM^3$
3	Sinusoidal	Regression	Miscellaneous	Y=a+b*cos(c*TEMP+d)

Universiti Utara Malaysia

Table 5.18 Continued

M	Name	Standard error	Correlation Coeff. (r)	Coeff.of Determination (r^2)	DOF	AICC
1	Sinusoidal	0.162446	0.733239	0.537640	24	-99.0911
2	Polynomial Deg.3	0.023078	0.995323	0.990668	24	-208.372
3	Sinusoidal	0.227579	0.304206	0.092541	24	-80.2106

5.4.4 TPS Effect on Knocking

With consideration of the fixed values of RPM and TEMP factors, the results shown that TPS is effective on knocking after apply the overall in formula in many situations to investigate its effect on Knocking. By observing the Figure 5.20 below, you will note the change in knocking when the values of TPS changes, with fixed temperature (Temp.) in the value of 89.5, and engine Revolution Per Minute (RPM) equals 5000, the value of the knocking will be equal 80. After increasing the (Temp) to 91.5, it is observed that the knocking is also increased, but when the (Temp) is continues rising, knock starts to decrease. Results from the application of the new model reflect low level knocking with increasing temperature (Temp) at the same points in Throttle (TPS), the Revolution Per Minute (RPM). This reflects the effectiveness of the new model with non-linear behaviour of the factors which affect the knock.

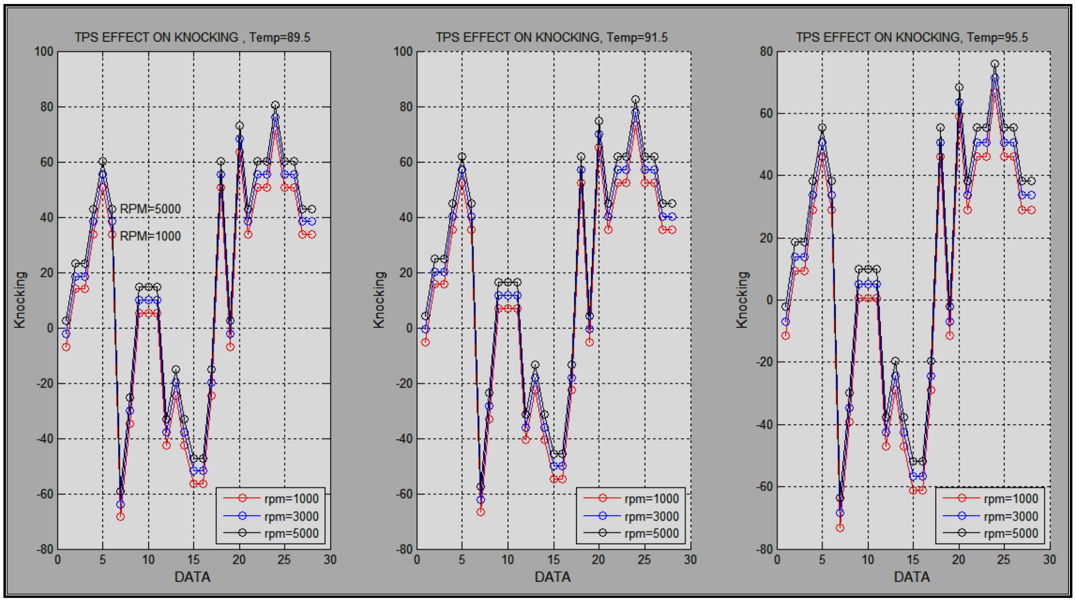


Figure 5.20. Effects TPS with different (Temp.) on knocking

Name	Value	Min	Max
MAX1	80.6768	80.6768	80.6768
MAX2	82.3962	82.3962	82.3962
MAX3	76.0238	76.0238	76.0238
MIN1	-59.0748	-59.07...	-59.07...
MIN2	-57.3554	-57.35...	-57.35...
MIN3	-63.7278	-63.72...	-63.72...
MODEL1	@(tps,rpm,temp)-0.2...		
Mean1	19.6465	19.6465	19.6465
Mean2	21.3659	21.3659	21.3659
Mean3	14.9935	14.9935	14.9935

UUM Universiti Utara Malaysia

By calculate the mean of knocking, we can get results more clearly as a Table 5.19.

Table 5.19

TPS effect on knocking behavior

CASES	Min_{kn}	Max_{kn}	$Mean_{kn}$
RPM=5000,TEMP=89.5	-59.0748	80.6768	19.6465
RPM=5000,TEMP=91.5	-57.3554	82.3962	21.3659
RPM=5000,TEMP=95.5	-63.7278	76.0238	14.9935

5.4.5 RPM Effect on Knocking

The results showed after the fixed each of (TPS) and (TEMP) the effect of (RPM) on the knocking with the change of TEMP in multiple cases, as in the following Figure 5.21:

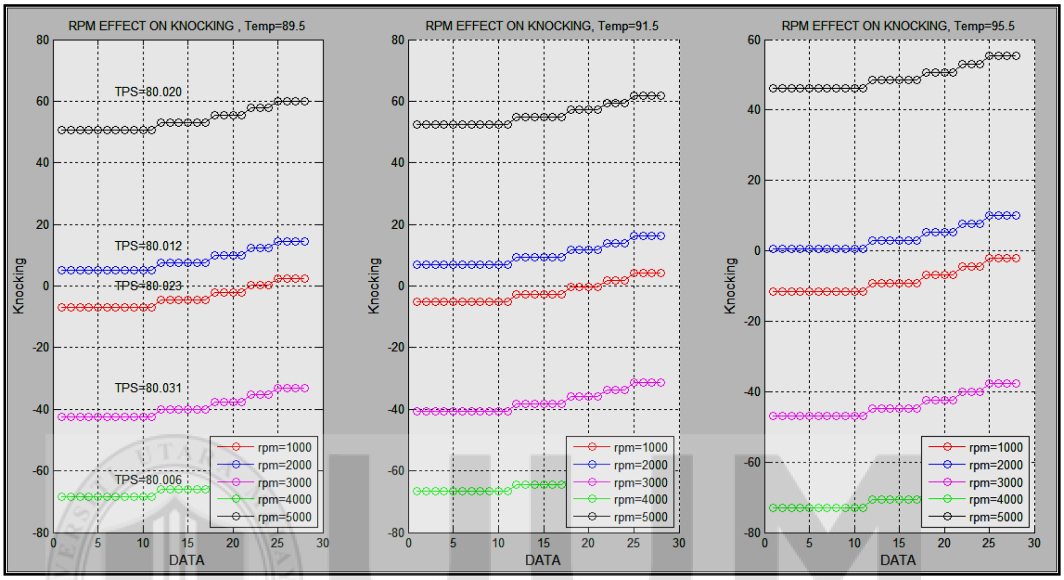


Figure 5.21. Effects RPM with different (Temp.) on knocking

Universiti Utara Malaysia

Name	Value	Min	Max
MAX1	2.5251	2.5251	2.5251
MAX2	4.2445	4.2445	4.2445
MAX3	-2.1279	-2.1279	-2.1279
MIN1	-6.8349	-6.8349	-6.8349
MIN2	-5.1155	-5.1155	-5.1155
MIN3	-11.4879	-11.48...	-11.48...
MODEL1	@(tps,rpm,temp)-0.2...		
Mean1	-3.5756	-3.5756	-3.5756
Mean2	-1.8563	-1.8563	-1.8563
Mean3	-8.2287	-8.2287	-8.2287

Table 5.20

RPM effect on knocking behavior

CASES	Min_{kn}	Max_{kn}	$Mean_{kn}$
TPS=80.023,TEMP=89.5	-6.8349	2.5251	-3.5756
TPS=80.023,TEMP=91.5	-5.1155	4.2445	-1.8563
TPS=80.023,TEMP=95.5	-11.48	-2.1279	-8.2287

For example, note in Figure 5.21 and the Table 5.20 decrease knocking, when the TEMP tend to change toward to height (TEMP=95.5, TPS=80.023 and RPM=1000), with consideration TEMP and TPS in the fixed in each case, which shows the effectiveness of partial its effect in the decreasing of knock and the work of the proposed model.

5.4.6 TEMP Effect on Knocking

Through the results get them shows us the active role of temperature (TEMP) in influencing the knocking. In Figure 5.22 and table 5.21, note with TEMP change and fixed (RPM) and (TPS=0.80.023) in each stage of the change in engine speed (RPM) decrease the level of knocking. This indicates that the partial effect of temperature as well as the effectiveness of the model in general, in reducing the level of knocking.

Table 5.21

TEMP effect on knocking behavior

CASES	Min_{kn}	Max_{kn}	$Mean_{kn}$
TPS=80.023,RPM=1000.5	-12.1383	-5.7543	-9.4449
TPS=80.023,RPM=2000.5	-9.7983	-3.4143	-7.1049
TPS=80.023,RPM=3000.5	-7.4583	-1.0743	-4.7649

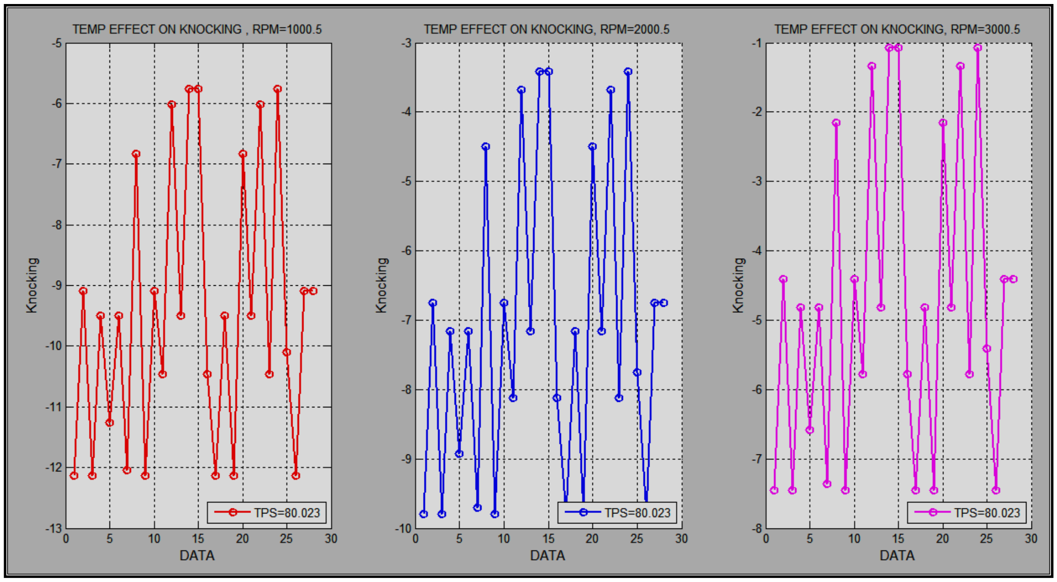


Figure 5.22. Effects TEMP with different (RPM) on knocking

Name	Value	Min	Max
MAX1	-5.7543	-5.7543	-5.7543
MAX2	-3.4143	-3.4143	-3.4143
MAX3	-1.0743	-1.0743	-1.0743
MIN1	-12.1383	-12.13...	-12.13...
MIN2	-9.7983	-9.7983	-9.7983
MIN3	-7.4583	-7.4583	-7.4583
MODEL1	@(tps,rpm,temp)-0.2...		
Mean1	-9.4449	-9.4449	-9.4449
Mean2	-7.1049	-7.1049	-7.1049
Mean3	-4.7649	-4.7649	-4.7649

UUM Universiti Utara Malaysia

If AIC score difference between two models is in magnitude of 1-2, the difference is significant.

Many executions have been applied and the results were as follows:

Proton_Turbo_Charge

1- 0.22562	2- 0.23819	3- 0.23863	4- 0.24663	5- 0.24663
6- 0.24663	7- 0.24663	8- 0.24663	9- 0.24691	10- 0.24693
11- 0.24691	12- 0.24691	13- 0.24691	14- 0.24691	15- 0.24691
16- 0.24904	17- 0.24904	18- 0.24869	19- 0.24869	20- 0.24773
21- 0.24815	22- 0.24815	23- 0.24815	24- 0.24815	

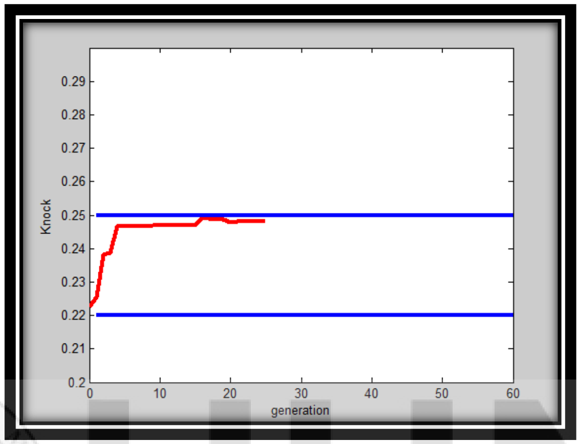
Elapsed time is 0.030314 seconds.

#generations=25 best Knock=0.24803

Best solution 80.013197 1218.1296 89.904968

Continuous genetic algorithm

Optimal_Factors = 80.013 1218.1 89.905



1	0.22272	2	0.22272	3	0.22272	4	0.23388	5	0.23388
6	0.2468	7	0.2468	8	0.2483	9	0.2483	10	0.2483
11	0.2483	12	0.2483	13	0.2483	14	0.2482	15	0.2482
16	0.2482	17	0.2482	18	0.2481	19	0.2481		

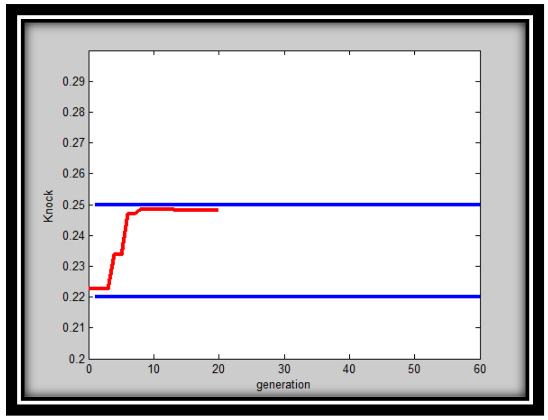
Elapsed time is 0.033244 seconds.

#generations=20 best Knock=0.24802

Best solution 80.023205 1199.9138 90.134682

Optimal_Factors = 80.023 1199.9 90.135

Universiti Utara Malaysia



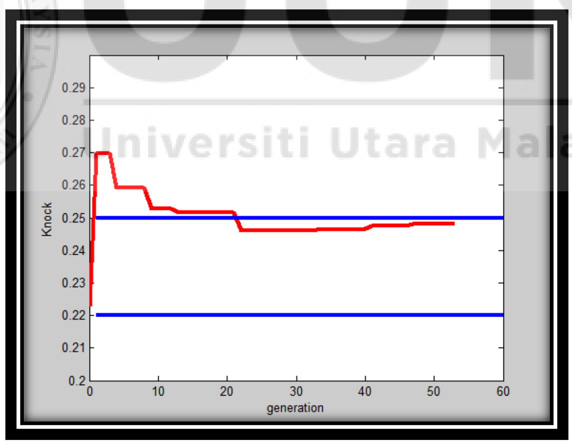
Elapsed time is 0.042071 seconds.

#generations=53 best Knock=0.24799

Best solution 80.021026 1205.459 91.36435

Optimal_Factors = 80.021 1205.5 91.364

UUM
Universiti Utara Malaysia

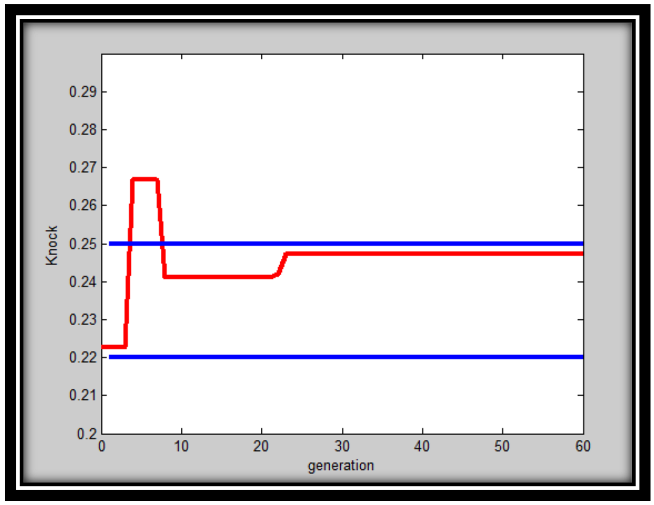


Elapsed time is 0.053894 seconds.

#generations=60 best Knock=0.24721

Best solution 80.012358 1215.3987 91.381835

Optimal_Factors = 80.012 1215.4 91.382



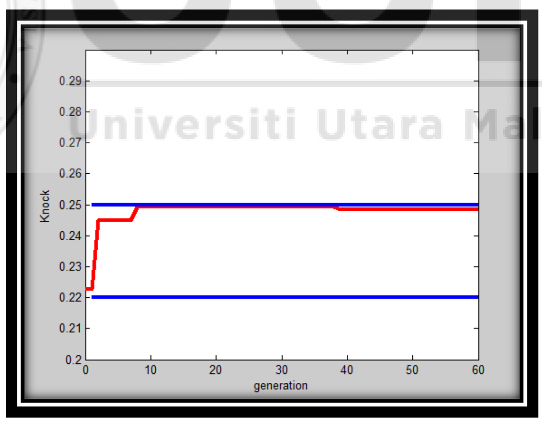
Elapsed time is 0.041634 seconds.

#generations=60 best Knock=0.24836

Best solution 80.344836 1195.6706 90.504088

Optimal_Factors = 80.345 1195.7 90.504

UUM
Universiti Utara Malaysia



EGMA_4FACTORS.xlxs file- Dodeg-charger

1. 0.0921 2. 0.0921 3. 0.1979 4. 0.1979 5. 0.1979 6. 0.1979 7. 0.1979

8. 0.2533 9. 0.2533 10. 0.2533 11. 0.2619 12. 0.2619 13. 0.2619 14. 0.2619

15. 0.2638 16. 0.2638 17. 0.2638 18. 0.2638 19. 0.2638 20. 0.2638 21. 0.2656

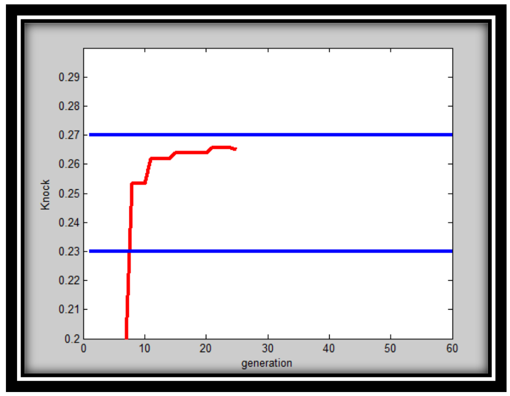
22. 0.2656 23. 0.2656 24. 0.2656

Elapsed time is 0.036748 seconds.

#generations=25 best Knock=0.26496

Best solution 7.8567836 1482.4319 53.378833

Optimal_Factors = 7.8568 1482.4 53.379

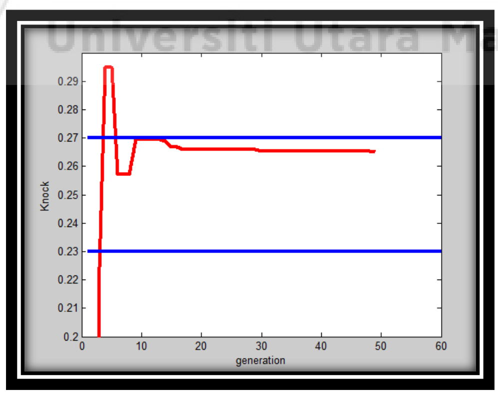


Elapsed time is 0.050063 seconds.

#generations=49 best Knock=0.26494

Best solution 7.5149151 2199.7697 60.492805

Optimal_Factors = 7.5149 2199.8 60.493



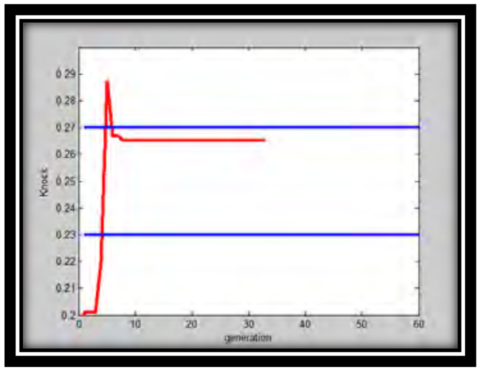
Elapsed time is 0.035558 seconds.

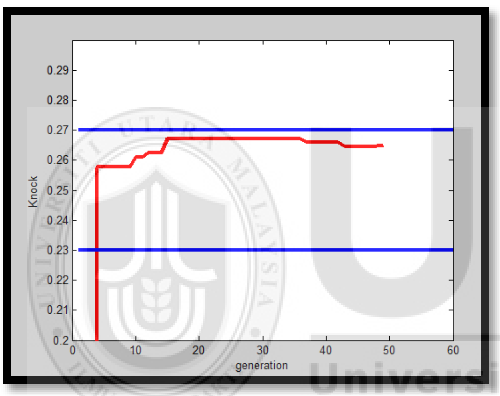
#generations=33 best Knock=0.26507

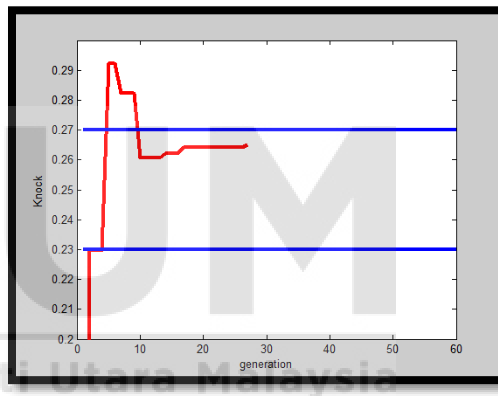
Best solution 7.5658721 2012.996 58.466991

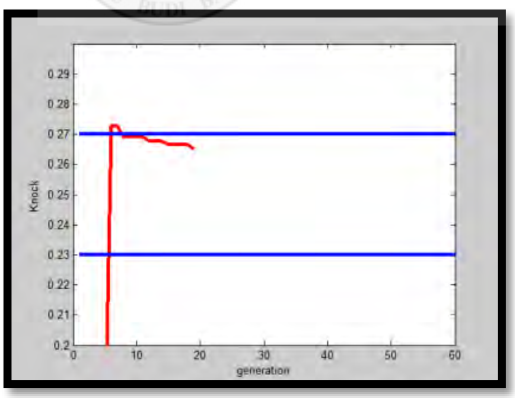
Optimal_Factors = 7.5659 2013 58.467

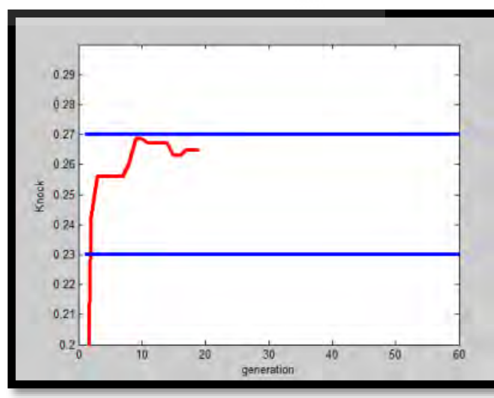
Universiti Utara Malaysia



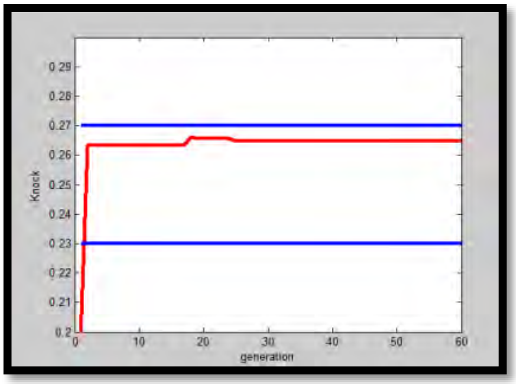


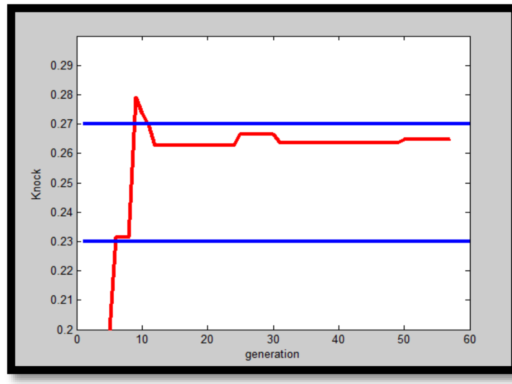


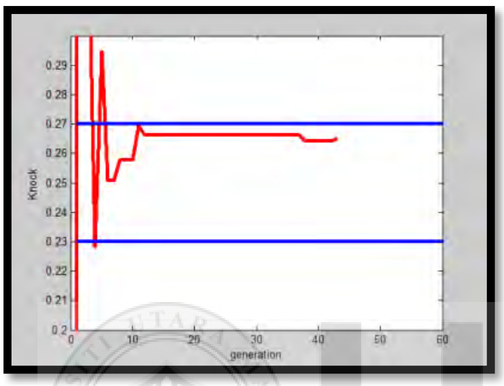


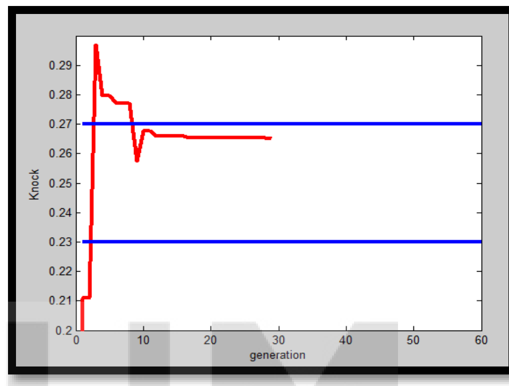


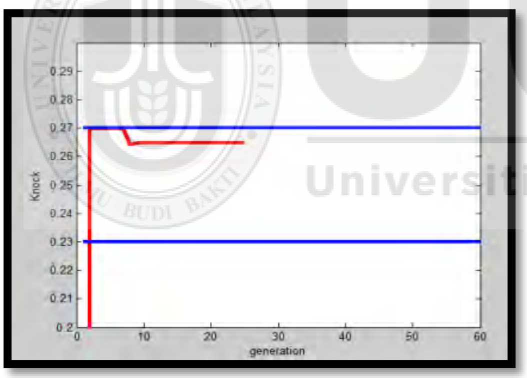
Universiti Utara Malaysia

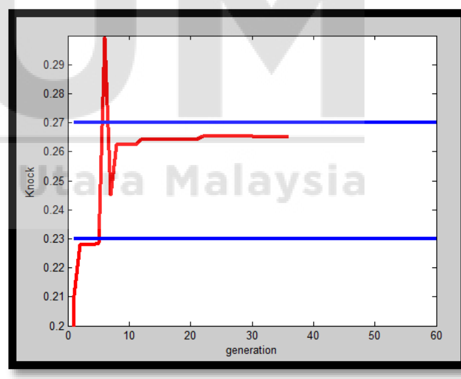


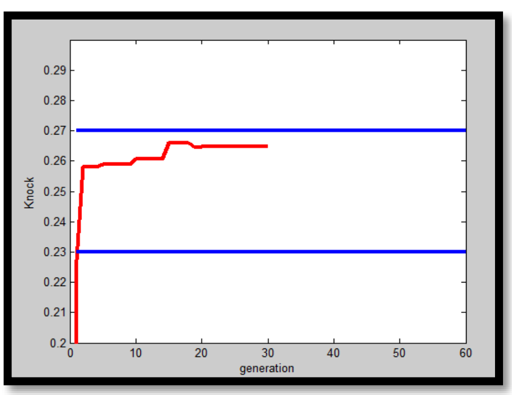


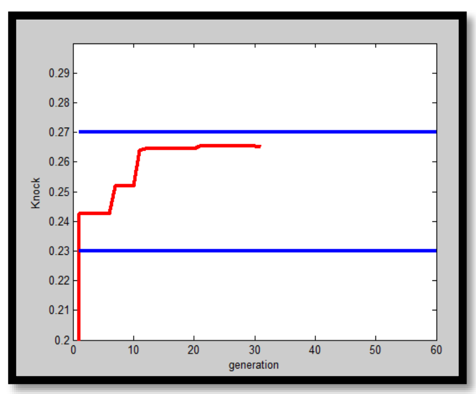




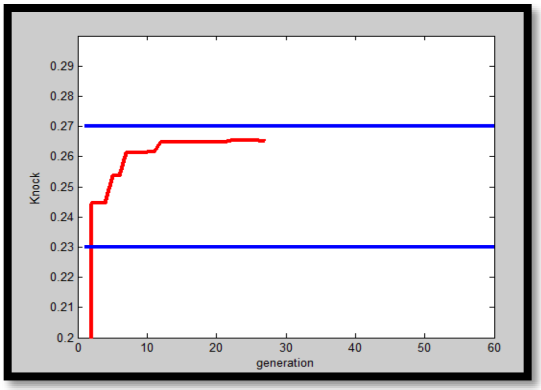


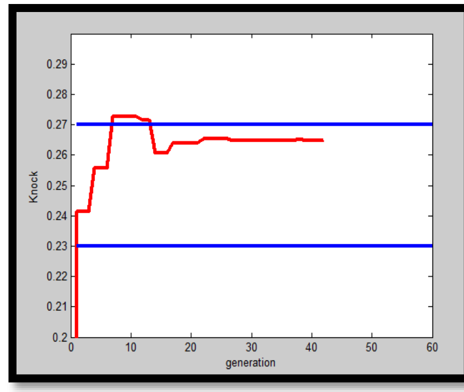


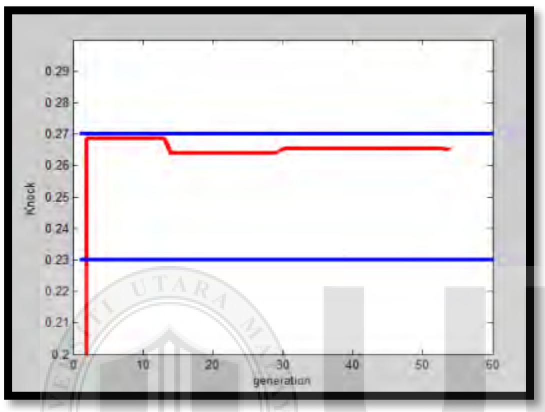


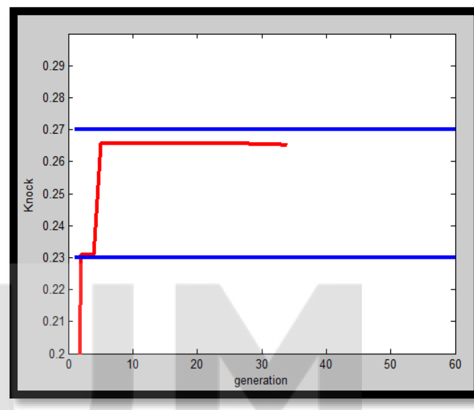


Universiti Utara Malaysia









UUM
Universiti Utara Malaysia

Hyundai_Genesis

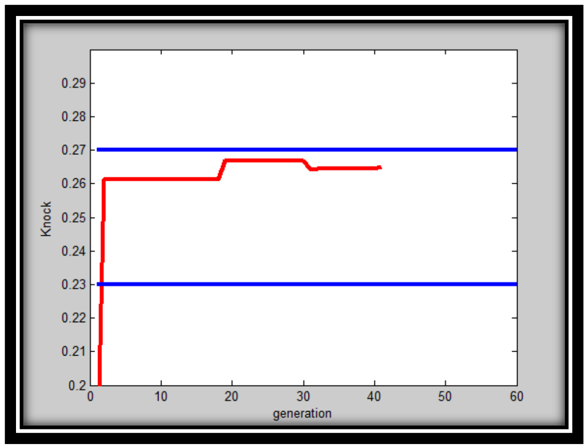
1	0.16552	2	0.26125	3	0.26125	4	0.26125	5	0.26125
6	0.26125	7	0.26125	8	0.26125	9	0.26125	10	0.26125
11	0.26125	12	0.26125	13	0.26125	14	0.26125	15	0.26125
16	0.26125	17	0.26125	18	0.26125	19	0.26685	20	0.26685
21	0.26685	22	0.26685	23	0.26685	24	0.26685	25	0.26685
26	0.26685	27	0.26685	28	0.26685	29	0.26685	30	0.26685
31	0.26422	32	0.26459	33	0.26459	34	0.26459	35	0.26459
36	0.26459	37	0.26459	38	0.26459	39	0.26462	40	0.26462

Elapsed time is 0.041626 seconds.

#generations=41 best Knock=0.26504

Best solution 7.646937 1776.0915 91.387088

Optimal_Factors = 7.6469 1776.1 91.387



UUM
Universiti Utara Malaysia

EGMA_KIA_Motors_Sorento

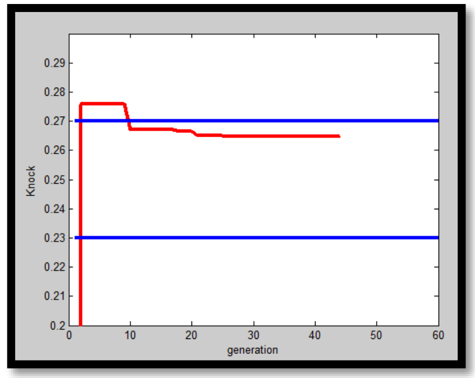
1	- 0.99678	2	0.27598	3	0.27598	4	0.27598	5	0.27598
6	0.27598	7	0.27598	8	0.27598	9	0.27598	10	0.26716
11	0.26716	12	0.26716	13	0.26716	14	0.26716	15	0.26716
16	0.26716	17	0.26702	18	0.2665	19	0.2665	20	0.2665
21	0.2652	22	0.2652	23	0.2652	24	0.2652	25	0.2652
26	0.2648	27	0.2648	28	0.2648	29	0.2648	30	0.264
31	0.2648	32	0.2648	33	0.2648	34	0.2648	35	0.2648
36	0.2648	37	0.26481	38	0.26481	39	0.26481	40	0.26481
41	0.26481	42	0.26481	43	0.26481				

Elapsed time is 0.046865 seconds.

#generations=44 best Knock=0.26499

Best solution 7.6713381 1717.5306 91.581631

Optimal_Factors = 7.6713 1717.5 91.582

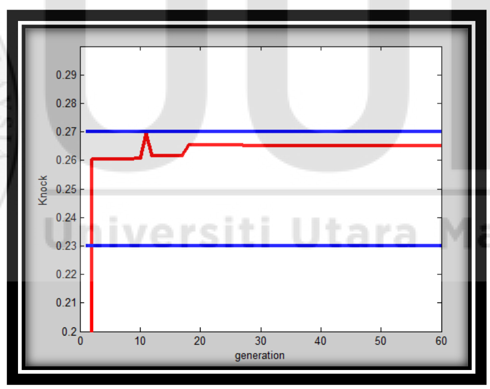


Elapsed time is 0.054338 seconds.

#generations=60 best Knock=0.26514

Best solution 7.5620406 2032.298 92.549749

Optimal_Factors = 7.562 2032.3 92.55



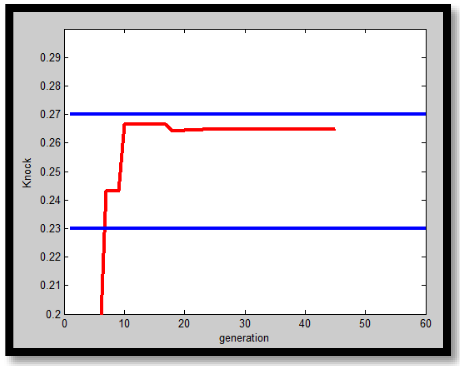
Elapsed time is 0.040090 seconds.

#generations=45 best Knock=0.26494

Best solution 7.4681168 2397.2582 89.914722

Optimal_Factors = 7.4681 2397.3 89.915

Universiti Utara Malaysia

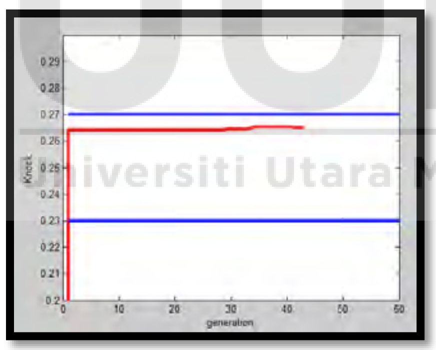


Elapsed time is 0.036770 seconds.

#generations=43 best Knock=0.26498

Best solution 7.4643629 2416.17 91.173984

Optimal_Factors = 7.4644 2416.2 91.174



5.5 Evaluation

In (2010), researchers Ismail and Yusof, referring to one of the methods to resolve the multi objectives optimization problems using the weighted sum approach, where the weights are generated randomly. Murata and Ishibuchi in (1995) and (1996) proposed an algorithm based on the weighted sum approach. This approached is called Random weight GA (RWGA). Their algorithm generates variable random weights vector at every GA iteration. By applying variable weighted-sum approached, various search directions are created in a single run without using additional parameters. This approach can produce a strong non-dominated solution that can be used as an initial solution for other techniques. However, this approach has difficulty in finding solutions which are uniformly distributed over nonconvex trade-off surface. The researcher claimed that he had obtained the better results from Schaffer algorithm VEGA in (1985).

After applying the same data for each two algorithms (Proposed algorithm (NWAEF) & Random Weights GA (RWGA), the results appear as showing in table 5.22.

Table

5.22 Evaluation two algorithms (NWAEF & RWGA)

Runs	**Proposed algorithm(NWAEF)**		**Random Weights GA(RWGA)**	
	Best knock	**Elapsed time**	**Best knock**	**Elapsed time**
1	0.2480	0.030314	0.2610	0.102232
2	0.2480	0.033244	0.2590	0.124155
3	0.2479	0.042071	0.2629	0.172331
4	0.2472	0.053894	0.2551	0.117408
5	0.2483	0.041634	0.2611	0.103411

Universiti Utara Malaysia

The findings, which appeared in the table above, shows that the value of the knock in the algorithm that generates random weights is the best of the proposed algorithm, relatively little variation in the results. But, on the other hand there is a relatively large variation in the time it takes to get to the best solution. Here we note that time is of great importance to reach the best solution.

5.6 Evaluation and Total Error Results

I've been compared to the behavior of the proposed model output with the output of the real model, see the following Figure 5.23, there has been showing relative error between the two models. The amount of error was calculated and the results were as follows:

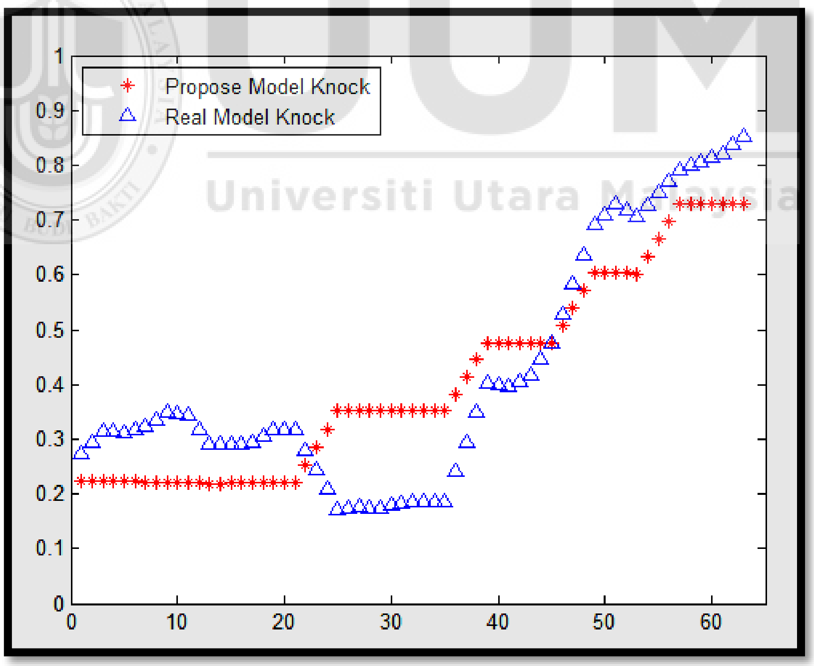


Figure 5.23. Total Error between Propose and Real Models

Difference between the actual data and calculated data is computed:

Def= Actual data- calculated data

Def= N_i-H_i

DS= $(N_i-H_i)^2$

SSE= $\sum$ DS

Hsize= sample size.

Total_Error = SSE/ Hsize.

Def = 0.0521 0.0721 0.0921 0.0912 0.0904 0.0949 0.0995 0.1130 0.1265
0.1242 0.1219 0.0977 0.0736 0.0721 0.0706 0.0716 0.0725 0.0855
0.0985 0.0974 0.0964 0.0275 -0.0413 -0.1101 -0.1788 -0.1771 -0.1755
-0.1759 -0.1763 -0.1716 -0.1669 -0.1662 -0.1655 -0.1652 -0.1649 -0.1418
-0.1187 -0.0956 -0.0725 -0.0763 -0.0800 -0.0699 -0.0598 -0.0310 -0.0021
0.0205 0.0432 0.0658 0.0885 0.1077 0.1268 0.1150 0.1031 0.0930
0.0829 0.0728 0.0627 0.0693 0.0758 0.0837 0.0916 0.1078 0.1240

Total_error = 0.0116

In order to evaluate the simulated model, the model has been compared with more than a model in which researchers have dealt with the phenomenon of knocking, through the factors affecting this phenomenon and calculate the total error of the model has been verified the effectiveness of the model in different conditions for each of the temperature, the speed of rotation of the engine, the ratio of air / fuel (λ).

The adoption of the temperature and pressure by researchers Douaud and Eyzat (1978), where it was fixed temperature coefficient in (X3=3800 K), while, the rest of the coefficients X1,X2 of the model are variable.

The model has been improved by the researchers Elmqvist et al. (2003), where it was fixed temperature coefficient and pressure coefficient X3,X2 respectively, while doing optimization for the third coefficient X1. Researchers adoption of the engine rotation speed (RPM) and air /fuel ratio (lambda) in the work of the optimization of the coefficient X3.

Results were presented by the following figure 5.24:

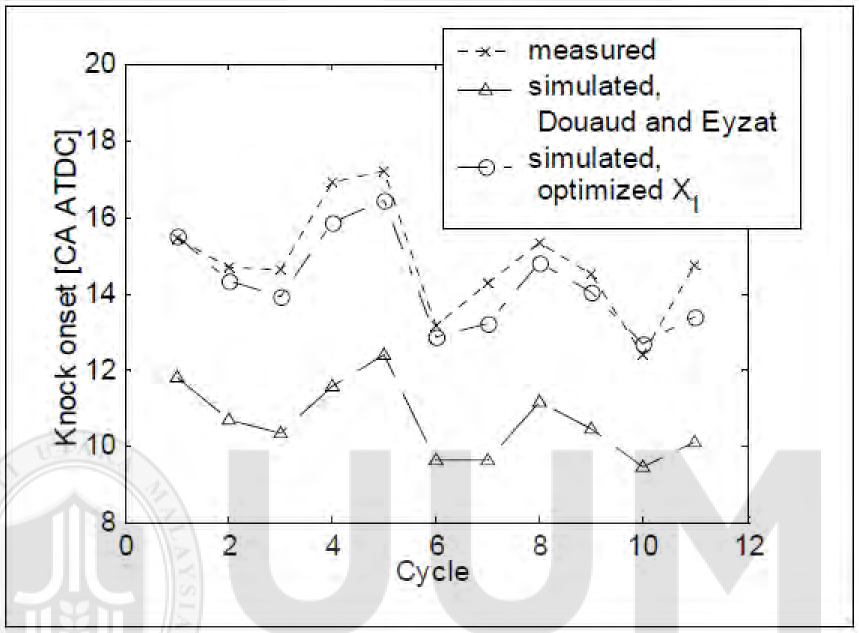


Figure 5.24. Comparison between model (Douaud & Eyzat) and (Elmqvist et al.)

According to the results in Table (5.23), researcher Elmqvist et al. (2003) shows the total amount of error of the model is:

Def= Actual data- calculated data

Def= N_i-H_i

DS= $(N_i-H_i)^2$

SSE= $\sum$ DS = 0.168

Hsize = Sample size = 11

Total_Error = SSE/ Hsize.

Total _Error = 0.168/11

Total_Error = 0.01527

Table 5.23

Difference average for (Elmqvist et al.) Model

Case [rpm]	2500	3000	3500
Measured average	14.7	13.05	8.6
Simulated average	14.5	12.85	8.9
Difference Average	0.16	0.2	-0.32

Figure 5.23 shows, according to the results that were obtained and have been previously calculated the overall error of the simulated model. We can say very good agreement between actual data (real) and simulation model of knock behavior.

In order to get on the case of the knock, the engine running in different situations, these cases can be found in Table 5.24 and Table 5.25.

Table 5. 24

Experimental cases(Elmqvist et al.)

Engine speed	2500, 3000, 3500 rpm
λ @ 2500 rpm	0.92
λ @ 3000 rpm	0.86, 0.99, 1.1
λ @ 3500 rpm	0.84
Fuel	95 RON
Coolant temperature	90°C

Table 5. 25

Experimental cases (Simulate Model)

Engine speed	1000, 2500, 3000, 3500 ,4000,4500, 5000 rpm
Tps @ 1000 rpm	80.012 – 80.032
Tps @ 2500 rpm	80.012 – 80.032
Tps @ 3000 rpm	80.012 – 80.032
Tps @ 3500 rpm	80.012 – 80.032
Tps @ 4000 rpm	80.012 – 80.032
Tps @ 4500 rpm	80.012 – 80.032
Tps @ 5000 rpm	80.012 – 80.032
Fuel	95 RON
Coolant temperature	89.5, 90, 90.6 °C

Universiti Utara Malaysia

Comparison between model Elmqvist et al. (2003) and simulation model illustrated in Table (5.26), where we note there is a difference in the amount of overall error for simulation model which shows the effectiveness of the model and the proximity of the real behavior of the knock.

Table 5.26

Error Comparison between model (Elmqvist et al.) and simulation model

	SSE	Hsize(Sample Size)	Total Error =SSE/Hsize
Elmqvist, C. Model	0.168	11	0.01527
Simulated Model	0.7337	28	0.0116
			0.00367

5.7 Locality of the problem

According to Rothlauf (2011) the locality of a problem describes how well the distances d(x,y) between any two solutions x,y ∈ X correspond to the difference of the objective values $| f(x) - f(y)|$. The locality of a problem is high if neighboring solutions have similar objective values, on the other hand, the locality of a problem is low if low distances do not correspond to low differences of the objective values.

Through the results shown in the tables 5.27,5.28,5.29,5.30 below, note that the difference (distance) between neighboring solutions is relatively slight. This reflects the robustness of the model.

Universiti Utara Malaysia

Table 5.27

Locality of a problem Proton_Turbo_Charge

NO.	Best Knock	Gen.	Best solution			Elapsed time
			TPS	RPM	TEMP.	
1	0.2480	25	80.013	1218.1	89.905	0.030314
2	0.2480	20	80.023	1199.9	90.135	0.033244
3	0.2479	53	80.021	1205.5	91.364	0.042071
4	0.2472	60	80.012	1215.4	91.382	0.053894
5	0.2483	60	80.345	1195.7	90.504	0.041634

Table 5.28

Locality of a problem EGMA-Dodeg-charger

NO.	Best Knock	Generation	Best solution			Elapsed time
			TPS	RPM	TEMP.	
1	0.26496	25	7.856	1482.4	53.379	0.03674
2	0.26494	49	7.514	2199.8	60.493	0.05006
3	0.26507	33	7.565	2013	58.467	0.03555
4	0.26506	54	7.866	1482.7	52.111	0.05949
5	0.26505	35	7.753	1572.9	67.678	0.03972

Universiti Utara Malaysia

Table 5.29

Locality of a problem EGMA-Hyundai_Genesis

NO.	Best Knock	Gen.	Best solution			Elapsed time
			TPS	RPM	TEMP.	
1	0.2650	41	7.646	1776.1	91.387	0.0416
2	0.2649	49	7.473	2374.8	91.333	0.0410
3	0.2650	53	7.498	2275.6	95.781	0.0455
4	0.2650	59	7.580	1967.1	91.279	0.0421
5	0.2637	60	7.451	2463	89.936	0.0482

Table 5.30

Locality of a problem EGMA-KIA_Motors_Sorento

NO.	Best Knock	Generation	Best solution			Elapsed time
			TPS	RPM	TEMP.	
1	0.26499	44	7.6713	1717.5	91.582	0.046865
2	0.26514	60	7.562	2032.3	92.55	0.054338
3	0.26494	45	7.4681	2397.3	89.915	0.040090
4	0.26498	43	7.4644	2416.2	91.174	0.036770
5	0.26495	56	7.4703	2390.4	91.674	0.052843

Universiti Utara Malaysia

5.8 Accuracy Results of Optimization

Table 5.31

Result of optimization Accuracy for 10 runs Dodeg engine.

Errors	Square_Error	Mean Square Error	Root_Mean_Square_Error
0.000029728	0.000000001	0.000000008	0.000090043
0.000056866	0.000000003		
0.000030426	0.000000001		
0.000040173	0.000000002	Sparkline Error level for 10 Runs	
0.000239930	0.000000058		
0.000086788	0.000000008		
0.000078157	0.000000006		
0.000020578	0.000000000		
0.000017045	0.000000000		
0.000049999	0.000000002		
Total	0.000000081		

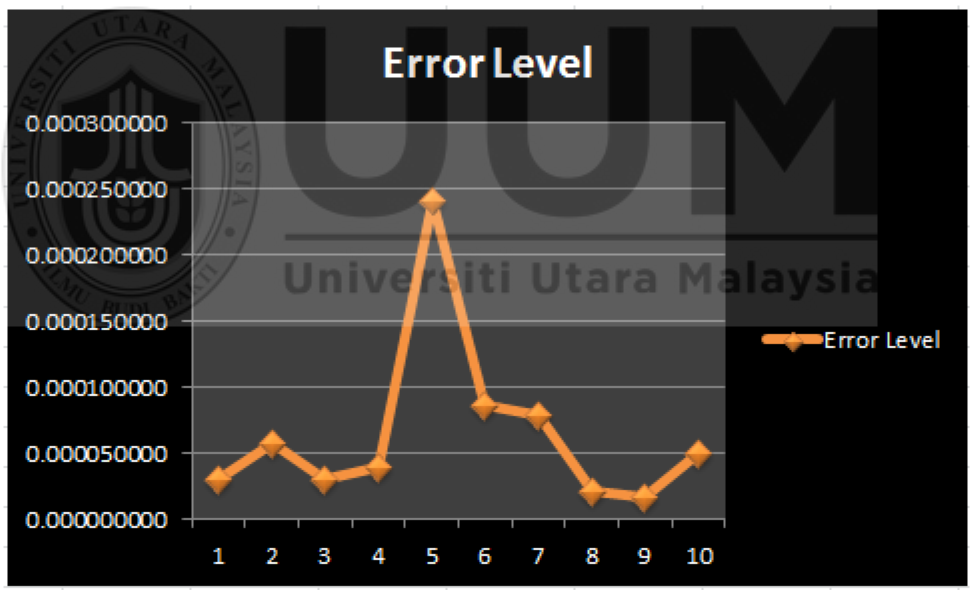
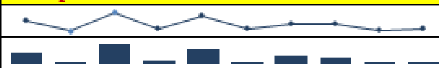


Table 5.32

Result of Optimization Accuracy for 10 runs Hyundai engine

Errors	Square_Error	Mean Square Error	Root_Mean_Square_Error
0.000099225	0.000000010	0.000000007	0.000080976
0.000007571	0.000000000		
0.000168460	0.000000028		
0.000030307	0.000000001	Sparkline Error level for 10 Runs	
0.000128610	0.000000017		
0.000024412	0.000000001		
0.000068587	0.000000005		
0.000061251	0.000000004		
0.000010390	0.000000000		
0.000025896	0.000000001		
Total	0.000000066		

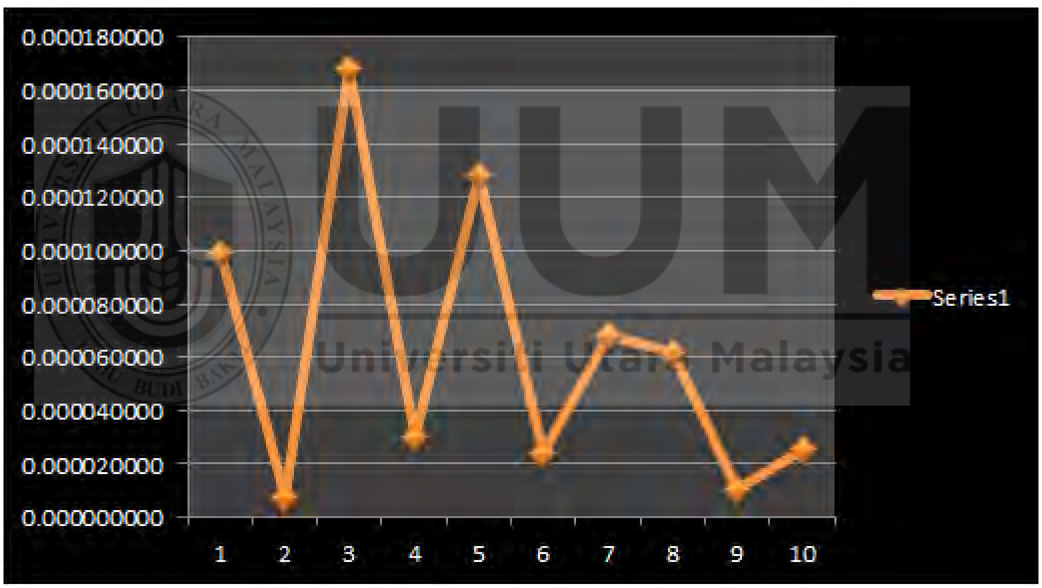
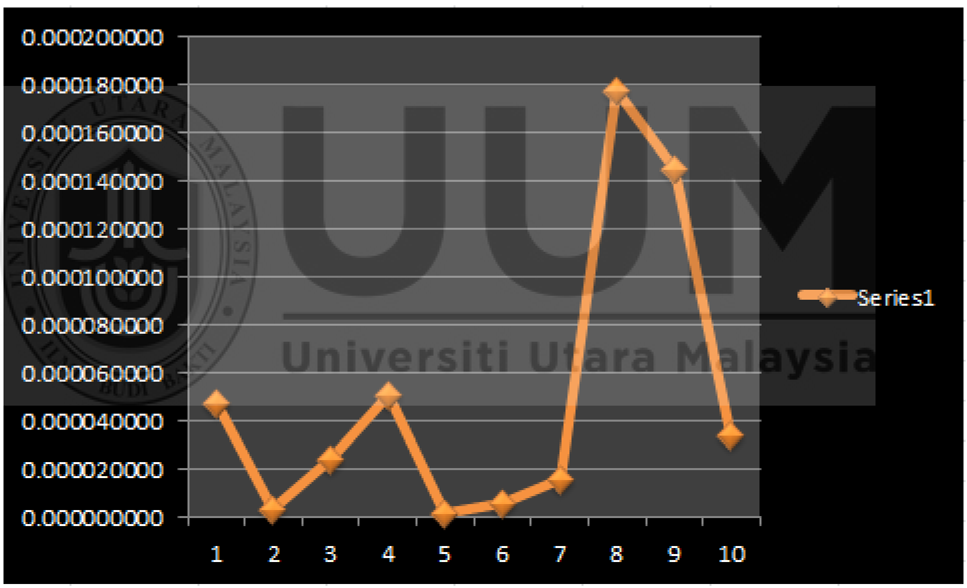


Table 5.33

Result of optimization Accuracy for 10 runs KIA engine

Errors	Square_Error	Mean Square Error	Root_Mean_Square_Error
0.000047713	0.000000002	0.000000006	0.000077126
0.000003214	0.000000000		
0.000024661	0.000000001		
0.000051161	0.000000003	Sparkline Error level for 10 Runs	
0.000001941	0.000000000		
0.000005841	0.000000000		
0.000016232	0.000000000		
0.000177390	0.000000031		
0.000145030	0.000000021		
0.000034192	0.000000001		
Total	0.000000059		

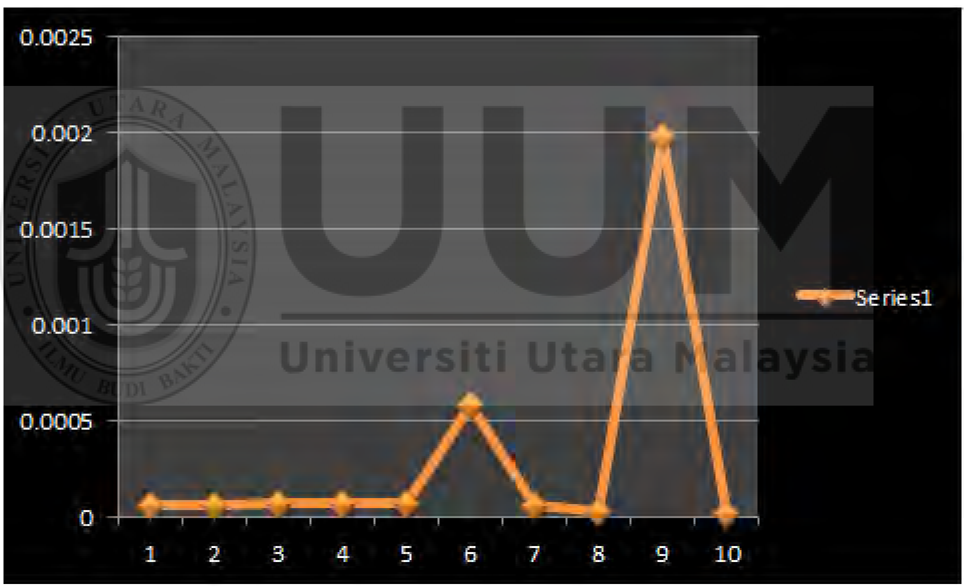


Universiti Utara Malaysia

Table 5.34

Result of optimization Accuracy for 10 runs proton engine

Errors	Square_Error	Mean Square Error	Root_Mean_Square_Error
0.000065097	0.000000004	0.000000431	0.000656762
0.000067071	0.000000004		
0.000078391	0.000000006		
0.000072707	0.000000005	**Sparkline Error level for 10 Runs**	
0.000078888	0.000000006		
0.00059002	0.000000348		
0.000068607	0.000000005		
0.000034355	0.000000001		
0.0019831	0.000003933		
0.000016688	0.000000000		
Total	0.000004313		



UUM
Universiti Utara Malaysia

5.9 Summary and Discussion

This chapter provides the results that were obtained according to the stages that were an explanation in the previous chapter (Chapter four).This include reasons analysis for choosing the three factors (Tps, Rpm,Temp), was dealt the reasons for selecting the one of three models. After that, in second part, analysis (objectives) in order to build the evaluation function (EF), in addition to show of the results that have been obtained to validate the model chosen. While, Part III, shows the results of the work of the proposed model for the evaluation function in the optimization problem between conflicting objectives, and obtain a form, for simulations knocking problem in internal combustion engines.

Once the correct model has been specified, relative importance weights allow for the interpretation of the model and the comparison of predictor variables. In the past, researchers have tended to focus on the rank order of predictors identified by the various importance analyses. However, this approach can be misleading, and we urge caution when examining the rank ordering of predictors. Another myth surrounding the application of relative importance analysis is that these weights solve the problem of multicolinearity among predictors. Although it is true that both dominance analysis and relative weight analysis were developed for use with correlated predictors and do in fact partition variance among correlated predictor variables, high levels of correlation among the predictor variables cannot be ignored. For example, if two or more predictor variables are very highly correlated because they are essentially tapping into the same underlying construct, the resulting importance weights can be misleading. One mistakenly held belief is that importance weights should be used to select predictors for a model. In actuality, both dominance analysis and relative weight analysis presuppose that the correct model has already been identified. For those interested in identifying the correct model, we recommend other procedures that may be more appropriate (e.g., procedures that help identify models with maximal R^2). If data are analyzed using ANOVA, and a significant F value obtained, a more detailed analysis of the differences between the treatment means will be required.

Universiti Utara Malaysia

CHAPTER SIX
CONCLUSION AND PERSPECTIVES

6.1 Introduction

In this chapter, Section 6.2 presents the general discussion of optimization algorithms and multi-objective optimization problems. In Section 6.3, the role of genetic algorithm (GA) in MOOPs is described. In section 6.4 involved the discussion of knock detection methods. Section 6.5 deals with achievement of research. The contributions of the research are clarified in Section 6.6. Limitations that are involved with this research are identified in Section 6.7. In Section 6.8, future works that are associated with this research are presented.

In general, the research addressed the limitation of the conflict among multiple objectives. Reducing the problem of determining the weights for each objective of the problem is also discussed. Many real-world problems of multiple objectives exist, which are difficult to deal with, moreover, it's not easy the relative importance of each objective is determined.

The main aim of the research is to reduce the complexity of evaluation function with conflicting objectives and to increase the speed of decision-making by reducing the calculations and by avoid determining the weights for each objective. High flexibility is also provided to add other objectives related to the solution of the problem.

Universiti Utara Malaysia

6.2 General Discussion

Because many contemporary problems are multi- objective, multi-objective optimization is of increasing importance for engineers and research science. Metaheuristic techniques are becoming a more popular approach at tackling such problems, most notably evolutionary algorithms.

After implementing the proposed algorithm for multi-objective optimization test functions, we conclude that the approach showed a good performance in converging with the true optimal solution. However, parameter setting in problems with higher number of variables is crucial.

Quantitative decisions in engineering, economics and science can be modelled using optimization tools and techniques. Ultimately, the decision maker tries to make the 'absolute best' decision possible that corresponds to the minimum (or maximum) suitable objective function, whilst satisfying a determined collection of feasibility constraints. An objective function will express an overall (modelled) system performance, examples including profit, utility, loss, risk or error. The present constraints may arise from technical, economic, physical or other considerations.

There may be overlap present amongst the optimization algorithms mentioned earlier. Also, it is possible to combine the search strategies, leading to a non-trivial issue in design search algorithms. Overall, robust optimization must find a trade-off between the quality of the solution and its robustness in terms of decision variable disturbance. It may be seen that the issue forms a multi-objective optimization problem. Not like optimization under uncertainty, the objective function in robust optimization may be deterministic.

6.3 Ability of GA to Solve MOOPs

GA is suitable for this class of problem and therefore considered a popular metaheuristic approach. Several methods allow traditional GAs to be customized to accommodate multi-objective problems, such as special fitness functions, methods to promote solution diversity, among others (Konak et al., 2006). GA is also suitable for solving multi-objective optimization problems because it is a population-based approach. The modification of a generic single-objective GA can lead to the determination of a set of multiple non-dominated solutions within a single run. GA is able to simultaneously search different regions of a solution area, which means finding diverse sets of solutions more possible for difficult problems with non-convex, discontinuous and multi-modal solution space. GA's cross over operator is able to exploit structures of good solutions with regards to different objectives in order to create a new non-dominated solution within unexplored areas of the Pareto front. Because of this, GA has become one of the most popular heuristic approaches to multi-objective design and optimization problems (Konak et al., 2006)

In (2002) state Jones et al. that 90% of approaches to multi-objective optimization aim to approximate the true Pareto front for the underlying problem. The majority of approaches utilize a metaheuristic technique, and 70% of all metaheuristic approaches were based on evolutionary approaches. Figure 6.1 shows that 70% of articles used GAs, which is a primary metaheuristic approach, 24% used simulated annealing, and 6% used Tabu search.

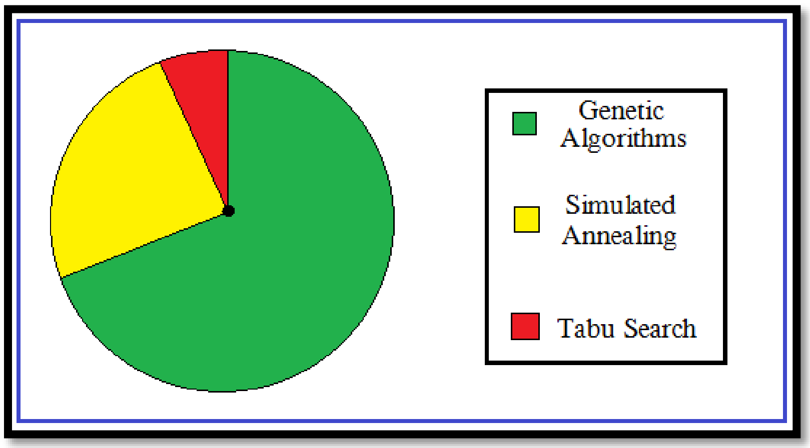


Figure 6.1. Breakdown of articles by primary metaheuristic methods

6.4 Knock Detection Methods Discussion

From the information of the methods to detect knocking, it is important to note that the statistical methods use Band Pass Filters (BPF) that rely on one or more of the resonant frequencies of the signal compression engine. When the speed of the rotation of the engine is slow these methods can be effective where the signal to noise ratio (SNR) is high. However, the signal-to-noise ratio becomes less important and is ineffective in the case of high speed rotation of the engine. This is in addition to the change in the resonance frequencies as a result of changing the engine compression ratio, and also changing the components of the fuel mixture (air/fuel), as well as the combustion of the mixture ratio and other reasons, such as the adoption of frequency representation or representation using time only.

Therefore, all of these restrictions cause these methods to have difficulty in providing exact measurements for knock density detection. In order to improve the efficiency of the detection accuracy, these difficulties and limitations have to be overcome through the use of analysis based on the representation of time - frequency, such as the Wigner distribution. However, this method may not be effective when the pressure signal (knocking) has two or more resonance frequencies in which case the Wigner distribution will suffer from a Cross-Term. There are many ways to get rid of the cross term by using a method called the Gabor-Wigner transform or Cohen's class distribution, but the large amount of computation required in these methods make them ineffective. In order to overcome this obstacle and improve the efficiency of detection accuracy it is possible to use fuzzy logic technology which features easy application, and requires only a short computation time. Moreover, it provides relevant information when a severe knock takes place, and in a comparison between fuzzy logic and conventional systems for knock detection, the fuzzy logic allows the determination of the non-linear correlation between inputs (knock factors) and outputs (knock intensity). Wavelet transform (WT) discrimination can be used in conjunction with fuzzy logic as fuzzy logic by itself has several limitations which could preclude its use in some cases, especially where the performance of fuzzy logic sometimes makes it difficult to analyze nonlinear effects. Fuzzy logic relies heavily on the experience of real life, which is a critical factor in the success of such control devices where a lack of experience could hinder the process.

6.5 Research achievement

The proposed main objective was achieved by construct a new Non-Weighted Aggregate Evaluation Function (NWAEF) for one of the Non-linear multi-objectives problems, which will simulate the knock behavior, in order to optimize non-linear decision factors (non-linear independent variables). All study objectives have been successfully achieved, which are summarized as follows.

- Objective 1: Select the factors (partial decision variables) that have strongest effect into decision-making problem, Construct and test three Non-linear regression models that having most influence on knocking by using ANOVA and some statistical tests. These models and analysis results in table 6.1 are :

  KNOCK = 988.455 - 12.3323 TPS - 0.0164546 TEMP + 0.00745664 IGN + 0.000100153 RPM

  KNOCK = 1001.73 - 12.4751 TPS - 0.0375599 TEMP + 0.000139913 RPM

  KNOCK = 929.707 - 11.6524 TPS + 0.0333036 TEMP + 0.0235528 IGN

Table 6.1

Summary analysis result for three Models

Model	Adj. R^2	Std. error of the estimation	F-Test	Sig. P-value	Durbin Watson
Model 1	0.884	0.0767	52.403	0.000	0.351
Model 2	0.877	0.0790	64.967	0.000	0.585
Model 3	0.820	0.0954	42.101	0.000	0.619

- Objective 2: Identify three optimal mathematical models that having most influence on knocking for system identification modeling by using curve fitting technique. These models are :

  Knock = a + b*cos(c*TPS + d) (Sinusoidal)

  knock= p + q*Rpm+ s*Rpm^2 (Polynomial second order)

  knock= m + n *cos(o*Temp + e) (sinusoidal)

- Objective 3: Individual objectives are successfully developed with aggregate partial derivatives method for each objective to prevent conflicting and mutual effect. The outcome for this process is:

Knock = -1*(0.24304*20.873* sin((29.873*tps-1405.39002) *pi/180)-1.08452 * 0.000117 * rpm-0.000009226 * 38.29344 * sin((13.29344 * temp + 35.15755) * pi/180)+5).

- Objective 4: In Optimize the single nonlinear multi-objective evaluation function and evaluate the non-weighted aggregate single nonlinear multi-objective evaluation function using genetic algorithm (GA) as a tool in terms of accuracy. The results are shown in Chapter 5 and appendix.

6.6 Contribution of the Research

The algorithm that overcomes the problem of determining the relative importance of mutual influence between multiple objectives and the main objective has been proposed by bypassing the problem of determining the weights for each objective problem. In internal combustion engine, the trade-off between performance and the risk of irreversible damages remains to be the key factor in the design of both low-consumption and high-performance engines (racing) (Cavina, Corti, Minelli, Moro, & Solieri, 2006).

Engine combustion control assumes a crucial role in reducing engine tailpipe emissions as well as maximizing performance. The amount of actuations influencing the combustion is on the rise, and as a consequence, control parameter calibrations (optimization) becomes challenging (Corti & Forte, 2011). In this research, several contributions may participate in dealing with a mathematical model (evaluation function), namely, multivariate nonlinear model. This model simulates a knock problem in internal combustion engines by accelerating the decision taken by electronic control unit (ECU), which provides the appropriate values (ideal) for decision-making. Knock is one of the key factors that affect performance and durability of spark ignition engines. It can also damage engine pistons. The objective of knock control is to provide an adjustment to the spark target based on the presence or absence of knock. Engine knock signal detection and control in internal combustion engines continues to be an important feature in engine management systems. The trade-off between performance and a risk of irreversible damage will remain a key factor in the design of both high-performance (racing) and low-consumption engines (Cavina et al., 2006).

6.7 Limitations

This research may have several limitations, such as difficulty of obtaining various devices to read data at the same time, because of their incompatibility with ECU and the diversity of the origin of the test engine data that were obtained.

6.8 Recommendations for Future work

Many primary and secondary factors influence knock process. No standard model exists for the representation of the knocking phenomenon. Thus, increasing the dimensions of the evaluation function (number of nonlinear factors) for building suitable and representation model for the process of knocking is possible through the foregoing view to build a multi-objective evaluation function MOEF that consists of three objectives.

The increase in evaluation function dimensions results in the possible reduction of evaluation function complexity through the construction and processing of the two factors in each of the objectives. Thus, computational time is reduced, and speed of decision-making submitted to other devices (actuators) is increased.

The use of fuzzy logic in chromosomes may contribute in solving the standardization (normalization) of problems in multi-objective optimization problems. The normalization problem (i.e., making the scores of the objectives comparable; often called standardization or normalization) Herwijnen (2011) is among the major weaknesses of aggregate objective functions. Multi-objective decision analysis requires the scores of various objectives to be transformed into comparable units. When the scales of the objectives are the same, their scores can be compared or combined.

6.9 Conclusion

Universiti Utara Malaysia

This work, a multi-disciplinary application in solving engineering problems was realized by adapting genetic algorithms to a constraint Non-weighted multi-objective optimization problem concerning an Engine Control Unit (ECU). After implementing the proposed algorithm for single objective optimization test functions, it was concluded that the approach showed a good performance in converging to the true optimal solution. However parameter setting in problems with higher number of variables is crucial.

By means of a case study, various genetic operators were adopted, showing that GAs emerges as being an applicable and promising approach to such complex engineering optimization problems. Moreover, the effects of different GA settings were depicted, leading to significant influences in respect of performance and convergence of GAs. For this purpose, knowledge from different fields was applied, including computer science, mathematical foundations of multi-objective optimization, automotive engineering, etc. In the course of the case study, an adaptable and configurable program was developed to automatically accomplish GAs with predefined settings to a sample wave engine model.

The program was utilized to perform various experiments, trying different settings. The results were analyzed in detail and several possible solutions were selected to be tested and evaluated, in order to confirm the model quality compared to the real engine and thus reliability of the optimization result. It is believed that the hybrid method may improve the ability of algorithm in finding the global optimal solutions. There is an essential need of comprehensive studies related to convergence of optimization problems, comparison metrics and different ways of combining single and multi-objective methods in order to conclude in a more precise and scientific manner.

Universiti Utara Malaysia

REFRENCES

Aarts, d. i. R. G. K. M. (2011). System Identification and Parameter Estimation (edition: 2011/2012 ed.). University of Twente / Faculty of Engineering Technology / Mechanical Automation and Mechatronics.

Adra, S. F., Griffin, I., & Fleming, P. J. (2009). A convergence acceleration technique for multiobjective optimisation *Multi-Objective Memetic Algorithms* (pp. 183-205): Springer.

Ahmad, H. A. (2012). The best candidates method for solving optimization problems. *Journal of Computer Science, 8*(5), 711.

Ahmadi, M. (2007). Intake, exhaust and valve timing design using single and multi-objective genetic algorithm: SAE Technical Paper.

Aho, K., Derryberry, D., & Peterson, T. (2014). Model selection for ecologists: the worldviews of AIC and BIC. *Ecology, 95*(3), 631-636.

Akaike, H. (1998). Information theory and an extension of the maximum likelihood principle *Selected Papers of Hirotugu Akaike* (pp. 199-213): Springer.

Al-Duwaish, H. (1997). *Control of nonlinear dynamical systems using genetic algorithms*.

Amouzgar., K. (2012). *Multi-Objective Optimization using Genetic Algorithms.*

Arora, J., Huang, M., & Hsieh, C. (1994). Methods for optimization of nonlinear problems with discrete variables: a review. *Structural optimization, 8*(2-3), 69-85.

Athan, T. W., & Papalambros, P. Y. (1996). A note on weighted criteria methods for compromise solutions in multi-objective optimization. *Engineering Optimization, 27*(2), 155-176.

Attar, A. A., & Karim, G. (1998). An analytical approach for the optimization of a SI engine performance including the consideration of knock: SAE Technical Paper.

Bandyopadhyay, S., & Saha, S. (2012). *Unsupervised classification: similarity measures, classical and metaheuristic approaches, and applications*: Springer Science & Business Media.

Bandyopadhyay, S., & Saha, S. (2013). Some single-and multiobjective optimization techniques *Unsupervised Classification* (pp. 17-58): Springer.

Beck, D. J., & Chaffin, D. (1992). An evaluation of inverse kinematics models for posture prediction. *Computer Applications in Ergonomics, Occupational safety and health*, 329-336.

Benson, H. P. (1995). Concave minimization: theory, applications and algorithms *Handbook of global optimization* (pp. 43-148): Springer.

Biegler, L. T. (2010). *Nonlinear programming: concepts, algorithms, and applications to chemical processes* (Vol. 10): SIAM.

Bonissone, P. P., Subbu, R., Eklund, N., & Kiehl, T. R. (2006). Evolutionary algorithms+ domain knowledge= real-world evolutionary computation. *Evolutionary Computation, IEEE Transactions on, 10*(3), 256-280.

Bozdogan, H. (2000). Akaike's information criterion and recent developments in information complexity. *Journal of mathematical psychology, 44*(1), 62-91.

Bozza, F., De Bellis, V., & Siano, D. (2014). A Knock Model for 1D Simulations Accounting for Cyclic Dispersion Phenomena: SAE Technical Paper.

Brecq, G., Bellettre, J., & Tazerout, M. (2003). A new indicator for knock detection in gas SI engines. *International Journal of Thermal Sciences, 42*(5), 523-532.

Caminiti, S., & Petreschi, R. (2005). String coding of trees with locality and heritability *Computing and Combinatorics* (pp. 251-262): Springer.

Carrejo, D. J., & Marshall, J. (2007). What is mathematical modelling? Exploring prospective teachers' use of experiments to connect mathematics to the study of motion. *Mathematics Education Research Journal, 19*(1), 45-76.

Cavina, N., Corti, E., Minelli, G., Moro, D., & Solieri, L. (2006). Knock indexes normalization methodologies: SAE Technical Paper.

Ching-Lai, H., & Abu, S. M. M. (1979). *Multiple Objective Decision Making, Methods and Applications: A State-of-the-art Survey*: Springer-Verlag.

Coello. (1996). An empirical study of evolutionary techniques for multiobjective optimization in engineering design.

Coello. (1999). A comprehensive survey of evolutionary-based multiobjective optimization techniques. *Knowledge and Information systems, 1*(3), 269-308.

Coello. (2000). *Handling preferences in evolutionary multiobjective optimization: A survey.* Paper presented at the Evolutionary Computation, 2000. Proceedings of the 2000 Congress on.

Coello. (2001). *A short tutorial on evolutionary multiobjective optimization.* Paper presented at the Evolutionary Multi-Criterion Optimization.

Coello, & Christiansen, A. D. (1998). Two new GA-based methods for multiobjective optimization. *Civil Engineering Systems, 15*(3), 207-243.

Coello, Lamont, G. B., & Van Veldhuizen, D. A. (2007). *Evolutionary algorithms for solving multi-objective problems*: Springer Science & Business Media.

Coello Coello, C. A. (2001). A Short Tutorial on Evolutionary Multiobjective Optimization.

Colby, M. K. (2013). *Theory and applications of difference evaluation functions.* Paper presented at the Proceedings of the 2013 international conference on Autonomous agents and multi-agent systems.

Corti, E., & Forte, C. (2011). *Real-Time Combustion Phase Optimization of a PFI Gasoline Engine*. http://dx.doi.org/10.4271/2011-01-1415

Cottrell, A. (2003). Regression analysis: basic concepts. *Available: Regression. pd*.

Dahleh, M. A. (2011). System Identification: MIT Lectures Notes 6.435-Lecture.

Dao, T. T. (2010). *Optimization And Control Of A Dynamical Process By Genetic Algorithm.* Paper presented at the ECMS.

Das, & Dennis, J. E. (1997). A closer look at drawbacks of minimizing weighted sums of objectives for Pareto set generation in multicriteria optimization problems. *Structural optimization, 14*(1), 63-69.

Das, & SENGUPTA, A. K. (1995). Technical note. Computer-aided human modelling programs for workstation design. *Ergonomics, 38*(9), 1958-1972.

Deb, K. (1999). *Solving goal programming problems using multi-objective genetic algorithms.* Paper presented at the Evolutionary Computation, 1999. CEC 99. Proceedings of the 1999 Congress on.

Deb, K., Agrawal, S., Pratap, A., & Meyarivan, T. (2000). A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. *Lecture notes in computer science, 1917*, 849-858.

Devijver, P. A., & Kittler, J. (1982). *Pattern recognition: A statistical approach* (Vol. 761): Prentice-Hall London.

di Gaeta, A., Giglio, V., Police, G., Reale, F., & Rispoli, N. (2010). Modeling Pressure Oscillations under Knocking Conditions: A Partial Differential Wave Equation Approach: SAE Technical Paper.

Dias, A. H., & De Vasconcelos, J. A. (2002). Multiobjective genetic algorithms applied to solve optimization problems. *Magnetics, IEEE Transactions on, 38*(2), 1133-1136.

Diener. (1995). Trajectory methods in global optimization *Handbook of Global optimization* (pp. 649-668): Springer.
Dill, K., Phillips, A., & Rosen, J. (1997). Molecular structure prediction by global optimization *Developments in Global Optimization* (pp. 217-234): Springer.
Dong, Z., Xu, J., Zou, N., & Chai, C. (2008). *Posture prediction based on orthogonal interactive genetic algorithm.* Paper presented at the Natural Computation, 2008. ICNC'08. Fourth International Conference on.
Douaud, A., & Eyzat, P. (1978). Four-octane-number method for predicting the anti-knock behavior of fuels and engines: SAE Technical Paper.
Draper, C. S. (1934). *The physical effects of detonation in a closed cylindrical chamber*: National Advisory Committee for Aeronautics.
Droste, S., & Wiesmann, D. (2003). On the design of problem-specific evolutionary algorithms *Advances in evolutionary computing* (pp. 153-173): Springer.
Dym, C. (2004). *Principles of mathematical modeling*: Academic press.
Elmqvist, C., Lindström, F., Ångström, H.-E., Grandin, B., & Kalghatgi, G. (2003). Optimizing engine concepts by using a simple model for knock prediction: SAE Technical Paper.
Eltaher, A. A. (2013). Localization of Engine Knock Events in Combustion Chambers Based on Mathematical-Physical Model of Motor Structure and Pressure Wave Propagation.
Faraway, J. J., Zhang, X., & Chaffin, D. B. (1999). Rectifying postures reconstructed from joint angles to meet constraints. *Journal of Biomechanics, 32*(7), 733-736.
Field, A. (2009). *Discovering statistics using SPSS*: Sage publications.
Floudas, C. A. (1999). *Deterministic global optimization* (Vol. 37): Springer.
Fonseca, C. M., & Fleming, P. J. (1993). *Genetic Algorithms for Multiobjective Optimization: FormulationDiscussion and Generalization.* Paper presented at the ICGA.
Franz, R. (2006). Representations for genetic and evolutionary algorithms: Springer–Verlag, Berling, Heidelberg, Netherlands.
Freschi, F., & Repetto, M. (2005). Multiobjective optimization by a modified artificial immune system algorithm *Artificial Immune Systems* (pp. 248-261): Springer.
Gabli, M., Jaara, E. M., & Mermri, E. B. (2014). A Genetic Algorithm Approach for an Equitable Treatment of Objective Functions in Multi-objective Optimization Problems. *IAENG International Journal of Computer Science, 41*(2).
Ganestam, P. (2010). *Empirical Knock Model for Automatic Engine Calibration.* M. Sc Thesis, Lund University.
Geisser, S. (1993). *Predictive inference* (Vol. 55): CRC Press.
Gen, & Cheng, R. (1997). Genetic algorithms and engineering design, 1997. *John Wily and Sons, New York*.
Gen, Ida, K., Li, Y., & Kubota, E. (1995). Solving bicriteria solid transportation problem with fuzzy numbers by a genetic algorithm. *Computers & Industrial Engineering, 29*(1), 537-541.
Glover, F., & Laguna, M. (1997). Tabu search, 1997. *Kluwer Academic Publishers*.
Golberg, D. E. (1989). Genetic algorithms in search, optimization, and machine learning. *Addion wesley, 1989*.
Gottlieb, J., Julstrom, B. A., Raidl, G. R., & Rothlauf, F. (2001). *Prüfer numbers: A poor representation of spanning trees for evolutionary search.* Paper presented at the Proceedings of the Genetic and Evolutionary Computation Conference.
Gottlieb, J., & Raidl, G. R. (2000). *The Effects of Locality on the Dynamics of Decoder-Based Evolutionary Search.* Paper presented at the GECCO.
Griffin, M. J. (2001). The validation of biodynamic models. *Clinical Biomechanics, 16*, S81-S92.

Grodzevich, O., & Romanko, O. (2006). Normalization and other topics in multi-objective optimization.

Haftka, R. T., & Gürdal, Z. (1992). *Elements of structural optimization* (Vol. 11): Springer Science & Business Media.

Hajela, P., & Lin, C.-Y. (1992). Genetic search strategies in multicriterion optimal design. *Structural optimization, 4*(2), 99-107.

Hansen, E. (1992). Global optimization using interval analysis, 1992. *Marcel Dekkar, New York, NY*.

Harishchandra, J., Ravindra, K., Sachin, B., & Thiyagarajan, S. (2010). Optimisation of Engine Cooling for Knock Suppression of Turbocharged Gasoline Engine.

Haupt, R. L., & Haupt, S. E. (2004). *Practical genetic algorithms*: John Wiley & Sons.

Herwijnen, M. v. (2011). Weighted Summation, www.ivm.vu.nl/en/Images/MCA2_tcm234-161528.pdf.

Hichert, J., Hoffmann, A., & Phú, H. X. (1997). Convergence speed of an integral method for computing the essential supremum *Developments in global Optimization* (pp. 153-170): Springer.

Horan, & Lavelle. (2005). Introduction to Partial Differentiation (Version 1.0 ed., pp. 30): PlymouthUniversity.

Horn, J. (1997). F1. 9 Multicriterion decision making. *Handbook of Evolutionary Computation, 97*(1).

Horn, J., Nafpliotis, N., & Goldberg, D. E. (1994). *A niched Pareto genetic algorithm for multiobjective optimization.* Paper presented at the Evolutionary Computation, 1994. IEEE World Congress on Computational Intelligence., Proceedings of the First IEEE Conference on.

Horst, & Romeijn. (2002). *Handbook of global optimization* (Vol. 2): Springer.

Horst, & Tuy. (1996a). *Global optimization: Deterministic approaches*: Springer.

Horst, & Tuy. (1996b). Global optimization: deterministic approaches. 1996. *Springer: Berlin). Horst, R. and Pardalos MP, Handbook of global optimization, l995 (Kluwer Academic Publishers: London). Ichida, K., Constrained optimization using interval analysis, Computers ind. Engng, 31*(3), 4.

Horst, R., & Romeijn, H. E. (2002). *Handbook of global optimization* (Vol. 2): Springer.

Howard, B., Cloutier, A., & Yang, J. J. (2012). Physics-based seated posture prediction for pregnant women and validation considering ground and seat pan contacts. *Journal of biomechanical engineering, 134*(7), 071004.

Hu, & Mehrotra, S. (2012). Robust and stochastically weighted multiobjective optimization models and reformulations. *Operations research, 60*(4), 936-953.

Hu, S. (2007). Akaike information criterion. *Center for Research in Scientific Computation*.

Hwang, C.-L., & Masud, A. S. M. (1979). *Multiple objective decision making—methods and applications*.

Igel, C. (1998). *Causality of hierarchical variable length representations.* Paper presented at the Evolutionary Computation Proceedings, 1998. IEEE World Congress on Computational Intelligence., The 1998 IEEE International Conference on.

Ismail, F. S., & Yusof, R. (2010). *A new self organizing multi-objective optimization method.* Paper presented at the Systems Man and Cybernetics (SMC), 2010 IEEE International Conference on.

Jakob, W., Gorges-Schleuter, M., & Blume, C. (1992). *Application of Genetic Algorithms to Task Planning and Learning.* Paper presented at the Parallel Problem Solving from Nature 2, PPSN-II, Brussels, Belgium, September 28-30, 1992.

Jaszkiewicz, A., Hapke, M., & Kominek, P. (2001). *Performance of multiple objective evolutionary algorithms on a distribution system design problem-Computational experiment.* Paper presented at the Evolutionary Multi-Criterion Optimization.

Jing, H. (2005). Local evaluation functions and global evaluation functions for computational evolution.

Jing, H., & Qingsheng, C. (2003). Emergence from local evaluation function. *Journal of Systems Science and Complexity, 16*(3), 372-390.

Joaquim, P., & Marques, S. (2007). Applied statistics using SPSS, statistica, Matlab and R. *Springer Company USA*, 205-211.

Jones, Brown, R. D., Clark, D. E., Willett, P., & Glen, R. C. (1993). *Searching databases of two-dimensional and three-dimensional chemical structures using genetic algorithms.* Paper presented at the Proceedings of the 5th International Conference on Genetic Algorithms.

Jones, Mirrazavi, S. K., & Tamiz, M. (2002). Multi-objective meta-heuristics: An overview of the current state-of-the-art. *European journal of operational research, 137*(1), 1-9.

Jubril, A. M. (2012). A nonlinear weights selection in weighted sum for convex multiobjective optimization. *Facta Universitatis, 27*(3), 357-372.

Jung, E. S., & Choe, J. (1996). Human reach posture prediction based on psychophysical discomfort. *International Journal of Industrial Ergonomics, 18*(2), 173-179.

Jung, E. S., & Park, S. (1994). Prediction of human reach posture using a neural network for ergonomic man models. *Computers & Industrial Engineering, 27*(1), 369-372.

Kasseris, E. P. (2011). *Knock limits in spark ignited direct injected engines using gasoline/ethanol blends.* Massachusetts Institute of Technology.

Kearfott. (1996). Rigorous Global Search: Continuous Problems Kluwer Academic Publishers. *Dordrecht, Netherlands*.

Khalil, S., Camal, R., & Laurent, T. (2009). Description of Knock Limit in a CFR Engine: Effects of Engine Settings and Gas Quality: SAE Technical Paper.

Khan, S. U. (2009). *A Self-adaptive Weighted Sum Technique for the Joint Optimization of Performance and Power Consumption in Data Centers.* Paper presented at the ISCA PDCCS.

Kim, Martin, B. J., & Gillespie, R. B. (2004). Modeling the coordinated movements of the head and hand using differential inverse kinematics: SAE Technical Paper.

Kim, & Weck, O. (2005). Adaptive weighted-sum method for bi-objective optimization: Pareto front generation. *Structural and multidisciplinary optimization, 29*(2), 149-158.

Kim, & Weck, O. (2006). Adaptive weighted sum method for multiobjective optimization: a new method for Pareto front generation. *Structural and multidisciplinary optimization, 31*(2), 105-116.

Knowles, J., & Hughes, E. J. (2005). *Multiobjective optimization on a budget of 250 evaluations.* Paper presented at the Evolutionary Multi-Criterion Optimization.

Kohavi, R. (1995). *A study of cross-validation and bootstrap for accuracy estimation and model selection.* Paper presented at the Ijcai.

Konak, A., Coit, D. W., & Smith, A. E. (2006). Multi-objective optimization using genetic algorithms: A tutorial. *Reliability Engineering & System Safety, 91*(9), 992-1007.

Kozarac, D., Tomic, R., Taritas, I., Chen, J.-Y., & Dibble, R. W. (2015). A Model for Prediction of Knock in the Cycle Simulation by Detail Characterization of Fuel and Temperature Stratification. *SAE International Journal of Engines, 8*(2015-01-1245).

Kuhn, H., & Tucker, A. (1951). Proceedings of the 2nd Berkeley Symposium on Mathematical Statistics and Probability: University of California Press.

Universiti Utara Malaysia

Levy, A. V., & Gómez, S. (1985). The tunneling method applied to global optimization. *Numerical optimization, 1981*, 213-244.

Liu, Begg, D., & Fishwick, R. (1998). Genetic approach to optimal topology/controller design of adaptive structures. *International Journal for Numerical Methods in Engineering, 41*(5), 815-830.

Liu, Li, W., Wu, T. B., Cheng, Y., Zhou, T. Y., & Zhu, G. F. (2014). *An Improved Constrained Optimization Multi-Objective Genetic Algorithm and its Application in Engineering.* Paper presented at the Advanced Materials Research.

Liu, G., & Chen, J. (2011). *The application of genetic algorithm based on matlab in function optimization.* Paper presented at the Electrical and Control Engineering (ICECE), 2011 International Conference on.

Ljung, L. (2010). Perspectives on system identification. *Annual Reviews in Control, 34*(1), 1-12.

Lochau, D.-I. M., Sun, B., Goltz, U., & Huhn, P. (2010). Model-based Parameter Optimization of an Engine Control Unit using Genetic Algorithms.

Lohmann, R. (1993). Structure evolution and incomplete induction. *Biological Cybernetics, 69*(4), 319-326.

Lonari, Y. (2011). Stochastic knock detection model for spark ignited engines.

Luke, S. (2015). Essentials of Metaheuristics. A Set of Undergraduate Lecture Notes. (Online Version 2.2). Department of Computer Science, George Mason University, October 2014. 261 p.

Luna, F., Nebro, A. J., & Alba, E. (2006). Parallel evolutionary multiobjective optimization *Parallel Evolutionary Computations* (pp. 33-56): Springer.

Ma, B. (2001). *Parametric and nonparametric approaches for multisensor data fusion.* The University of Michigan.

Maier, M. W., & Rechtin, E. (2000). *The art of systems architecting* (Vol. 2): CRC press Boca Raton.

Marler, R. T., & Arora, J. S. (2010). The weighted sum method for multi-objective optimization: new insights. *Structural and multidisciplinary optimization, 41*(6), 853-862.

Merola, S., Sementa, P., & Tornatore, C. (2011). Experiments on knocking and abnormal combustion through optical diagnostics in a boosted spark ignition port fuel injection engine. *International journal of automotive technology, 12*(1), 93-101.

Messac, A., & Mattson, C. A. (2002). Generating well-distributed sets of Pareto points for engineering design using physical programming. *Optimization and Engineering, 3*(4), 431-450.

Michalewicz, Z. (1996). *Genetic algorithms+ data structures= evolution programs*: springer.

Michalewicz, Z., Schmidt, M., Michalewicz, M., & Chiriac, C. (2005). *Evaluation functions for real world problems: A case study.* Paper presented at the Proceedings of the 16th Mini–EURO Conference and 10th Meeting of EWGT.

Millo, F., Rolando, L., Pautasso, E., & Servetto, E. (2014). A Methodology to Mimic Cycle to Cycle Variations and to Predict Knock Occurrence through Numerical Simulation: SAE Technical Paper.

Mockus, J. (2002). Bayesian heuristic approach to global optimization and examples. *Journal of Global Optimization, 22*(1-4), 191-203.

Mockus, J. (2010). *Bayesian Heuristic approach to discrete and global optimization: Algorithms, visualization, software, and applications*: Springer-Verlag.

Mockus, J., Eddy, W., Mockus, A., Mockus, L., & Reklaitis, G. (1996). Bayesian Heuristic Approach to Discrete and Global Optimization: Algorithms, Visualization. *Software, and Applications. Kluwer Academic Publishers, Dordrecht*.

Universiti Utara Malaysia

Moré, J. J., & Wu, Z. (1997). Global continuation for distance geometry problems. *SIAM Journal on Optimization, 7*(3), 814-836.

Mosteller, F., & Doob, L. W. (1949). *The pre-election polls of 1948*: Social Science Research Council.

Murata, T., & Ishibuchi, H. (1995). *MOGA: Multi-objective genetic algorithms.* Paper presented at the Evolutionary Computation, 1995., IEEE International Conference on.

Murata, T., Ishibuchi, H., & Tanaka, H. (1996). Multi-objective genetic algorithm and its applications to flowshop scheduling. *Computers & Industrial Engineering, 30*(4), 957-968.

Nedjah, N., & de Macedo Mourelle, L. (2005). *Real-world multi-objective system engineering*: Nova Publishers.

Neumaier. (1990). *Interval methods for systems of equations* (Vol. 37): Cambridge university press.

Nowak, R. D. (2002). Nonlinear system identification. *Circuits, Systems and Signal Processing, 21*(1), 109-122.

Osman, I. H., & Kelly, J. P. (1996). *Meta-heuristics: theory and applications*: Springer.

Paduart, J. (2008). *Identification of Nonlinear Systems using Polynomial Nonlinear State Space Models.*

Paduart, J., Lauwers, L., Swevers, J., Smolders, K., Schoukens, J., & Pintelon, R. (2010). Identification of nonlinear systems using polynomial nonlinear state space models. *Automatica, 46*(4), 647-656.

Paulden, T., & Smith, D. K. (2006). From the Dandelion Code to the Rainbow Code: A class of bijective spanning tree representations with linear complexity and bounded locality. *Evolutionary Computation, IEEE Transactions on, 10*(2), 108-123.

Peyton. (2014). New Approaches to Knock Simulation, Analysis, and Control. [Presentation].

Peyton Jones, J. C., Spelina, J. M., & Frey, J. (2013). Likelihood-based control of engine knock. *Control Systems Technology, IEEE Transactions on, 21*(6), 2169-2180.

Pham, Q., & Coulter, S. (1995). *Modelling the chilling of pig carcasses using an evolutionary method.* Paper presented at the Proc. Int. Congress of Refrig.

Pintér. (1996a). Global optimization in action. Continuous and Lipschitz optimization: Algorithms, implementation and applications: Kluwer, Dordrecht.

Pintér. (1996b). *Global optimization in action: continuous and Lipschitz optimization: algorithms, implementations and applications* (Vol. 6): Springer.

Pro, I. (2007). Version 6.0, WaveMetrics: Inc.

Puchta, M., & Gottlieb, J. (2002). Solving car sequencing problems by local optimization *Applications of Evolutionary Computing* (pp. 132-142): Springer.

Rachmawati, L., & Srinivasan, D. (2006). *A multi-objective genetic algorithm with controllable convergence on knee regions.* Paper presented at the Evolutionary Computation, 2006. CEC 2006. IEEE Congress on.

Raidl, G. R., & Gottlieb, J. (2005). Empirical analysis of locality, heritability and heuristic bias in evolutionary algorithms: A case study for the multidimensional knapsack problem. *Evolutionary computation, 13*(4), 441-475.

Ratschek, H., & Rokne, J. (1988). *New computer methods for global optimization*: Halsted Press.

Rechenberg, I. (1994). Evolutionsstrategie'94, volume 1 of Werkstatt Bionik und Evolutionstechnik. *Frommann {Holzboog, Stuttgart*.

Reeves, C. R. (1995). Modern heuristic techniques for combinatorial problems. 1995: McGraw-Hill International Limited UK.

Revier, B. M. (2006). *Phenomena that determine knock onset in spark-ignited engines.* Massachusetts Institute of Technology.

Richardson, J. T., Palmer, M. R., Liepins, G. E., & Hilliard, M. (1989). *Some guidelines for genetic algorithms with penalty functions.* Paper presented at the Proceedings of the third international conference on Genetic algorithms.

Rothlauf, F. (2011). *Design of modern heuristics: principles and application*: Springer.

Rothlauf, F., & Goldberg, D. (1999). *Tree network design with genetic algorithms-an investigation in the locality of the prüfernumber encoding.* Paper presented at the Late Breaking Papers at the Genetic and Evolutionary Computation Conference.

Ryu, J.-h., Kim, S., & Wan, H. (2009). *Pareto front approximation with adaptive weighted sum method in multiobjective simulation optimization.* Paper presented at the Winter Simulation Conference.

Saaty, T. L., & Vargas, L. (1991). The Logic of Priorities. Applications of the Analytic Hierarchy Process in Business, Energy, Health & Transportation, Vol. III of the AHP Series: RWS Publications, Pittsburgh.

Santana-Quintero, L. V., Montano, A. A., & Coello, C. A. C. (2010). A review of techniques for handling expensive functions in evolutionary multi-objective optimization *Computational intelligence in expensive optimization problems* (pp. 29-59): Springer.

Sarker, R. A., & Newton, C. S. (2007). *Optimization modelling: a practical approach*: CRC Press.

Schaffer, J. D. (1985). *Multiple objective optimization with vector evaluated genetic algorithms.* Paper presented at the Proceedings of the 1st international Conference on Genetic Algorithms.

Schneeweiss, C. (2003). *Distributed decision making*: Springer.

Sendhoff, B., Kreutz, M., & Von Seelen, W. (1997). A condition for the genotype-phenotype mapping: Causality. *arXiv preprint adap-org/9711001*.

Shan, S., & Wang, G. G. (2005). An efficient Pareto set identification approach for multiobjective optimization on black-box functions. *Journal of Mechanical Design, 127*(5), 866-874.

Sheng, W., Swift, S., Zhang, L., & Liu, X. (2005). A weighted sum validity function for clustering with a hybrid niching genetic algorithm. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on, 35*(6), 1156-1167.

Shih, S., Itano, E., Xin, J., Kawamoto, M., & Maeda, Y. (2003). Engine knock toughness improvement through water jacket optimization: SAE Technical Paper.

Sindhya, K. (2011). *Hybrid evolutionary multi-objective optimization with enhanced convergence and diversity.* PhD thesis, Department of Mathematical Information Technology, University of Jyväskylä, Finland.

Spelina, J. M., Jones, J. C. P., & Frey, J. (2014). Recent Advances in Knock Analysis, Simulation, and Control: SAE Technical Paper.

Srinivas, N., & Deb, K. (1994). Muiltiobjective optimization using nondominated sorting in genetic algorithms. *Evolutionary computation, 2*(3), 221-248.

Strongin, R. G., & Sergeyev, Y. D. (2000). *Global optimization with non-convex constraints: Sequential and parallel algorithms* (Vol. 45): Springer.

Sykes, A. O. (1993). An introduction to regression analysis.

Syswerda, G., & Palmucci, J. (1991). *The Application of Genetic Algorithms to Resource Scheduling.* Paper presented at the ICGA.

Taglialatela, F., Moselli, G., & Lavorgna, M. (2005). Engine knock detection and control using in-cylinder pressure signal and soft computing techniques: SAE Technical Paper.

Talbi. (2009). *Metaheuristics: from design to implementation* (Vol. 74): John Wiley & Sons.

Universiti Utara Malaysia

Talbi, Basseur, M., Nebro, A. J., & Alba, E. (2012). Multi-objective optimization using metaheuristics: non-standard algorithms. *International Transactions in Operational Research, 19*(1-2), 283-305.

Tang, W., Cavazza, M., Mountain, D., & Earnshaw, R. (1999). A constrained inverse kinematics technique for real-time motion capture animation. *The Visual Computer, 15*(7), 413-425.

Tappeta, R., Renaud, J., & Rodríguez, J. (2002). An interactive multiobjective optimization design strategy for decision based multidisciplinary design. *Engineering Optimization, 34*(5), 523-544.

Tarun, P. K. (2008). *A dynamic multiple stage, multiple objective optimization model with an application to a wastewater treatment system*: ProQuest.

Thomas, C. (1996). Introduction to Differential Calculus.

Thomasson, A., Eriksson, L., Lindell, T., Peyton Jones, J. C., Spelina, J., & Frey, J. (2013). *Tuning and experimental evaluation of a likelihood-based engine knock controller.* Paper presented at the Decision and Control (CDC), 2013 IEEE 52nd Annual Conference on.

Tolani, D., Goswami, A., & Badler, N. I. (2000). Real-time inverse kinematics techniques for anthropomorphic limbs. *Graphical models, 62*(5), 353-388.

Tran, L.-N., Hanif, M. F., Tölli, A., & Juntti, M. (2012). Fast converging algorithm for weighted sum rate maximization in multicell MISO downlink. *Signal Processing Letters, IEEE, 19*(12), 872-875.

Trummer, I., & Koch, C. (2014). *Approximation schemes for many-objective query optimization.* Paper presented at the Proceedings of the 2014 ACM SIGMOD international conference on Management of data.

Vancoillie, J., Sileghem, L., & Verhelst, S. (2013). Development and validation of a knock prediction model for methanol-fuelled SI engines: SAE Technical Paper.

Voss, S., Osman, I. H., & Roucairol, C. (1999). *Meta-heuristics: Advances and trends in local search paradigms for optimization*: Kluwer Academic Publishers.

Vossoughi, & Rezazadeh, S. (2004). Development of an Integrated Computer Environment for Optimization of Engine Management System Calibration: SAE Technical Paper.

Vossoughi, & Rezazadeh, S. (2005). *Optimization of the calibration for an internal combustion engine management system using multi-objective genetic algorithms.* Paper presented at the Evolutionary Computation, 2005. The 2005 IEEE Congress on.

Wang, X. (1999). A behavior-based inverse kinematics algorithm to predict arm prehension postures for computer-aided ergonomic evaluation. *Journal of Biomechanics, 32*(5), 453-460.

Wang, X., & Verriest, J. P. (1998). A geometric algorithm to predict the arm reach posture for computer-aided ergonomic evaluation. *The journal of visualization and computer animation, 9*(1), 33-47.

Weicker, K., & Weicker, N. (1999). Locality vs. randomness–dependence of operator quality on the search state. *Foundations of Genetic Algorithms, 5*, 147-163.

Whitley, D., & Rowe, J. E. (2005). Gray, binary and real valued encodings: Quad search and locality proofs *Foundations of Genetic Algorithms* (pp. 21-36): Springer.

Wienke, D., Lucasius, C., & Kateman, G. (1992). Multicriteria target vector optimization of analytical procedures using a genetic algorithm: Part I. theory, numerical simulations and application to atomic emission spectroscopy. *Analytica Chimica Acta, 265*(2), 211-225.

Williams, B., Brown, T., & Onsman, A. (2012). Exploratory factor analysis: A five-step guide for novices. *Australasian Journal of Paramedicine, 8*(3), 1.

Universiti Utara Malaysia

Wilson, Cappelleri, D., Simpson, T. W., & Frecker, M. (2001). Efficient Pareto frontier exploration using surrogate approximations. *Optimization and Engineering, 2*(1), 31-50.

Wilson, & Macleod, M. (1993). *Low implementation cost IIR digital filter design using genetic algorithms.* Paper presented at the IEE/IEEE workshop on natural algorithms in signal processing.

Yang. (2011). Bat algorithm for multi-objective optimisation. *International Journal of Bio-Inspired Computation, 3*(5), 267-274.

Yang, Karamanoglu, M., & He, X. (2013). Multi-objective flower algorithm for optimization. *Procedia Computer Science, 18*, 861-868.

Yang, Marler, R. T., Kim, H., Arora, J., & Abdel-Malek, K. (2004). *Multi-objective optimization for upper body posture prediction.* Paper presented at the 10th AIAA/ISSMO multidisciplinary analysis and optimization conference.

Zadbood, A., & Noghondarian, K. (2012). Multiple Response Surface Optimization Considering the Decision Maker's Preference Information. *Advanced Materials Research, 433*, 1646-1652.

Zebulum, R. S., Pacheco, M. A., & Vellasco, M. (1998). *A multi-objective optimisation methodology applied to the synthesis of low-power operational amplifiers.* Paper presented at the In Ivan Jorge Cheuri and Carlos Alberto dos Reis Filho, editors, Proceedings of the XIII International Conference in Microelectronics and Packaging.

Zhang, Domaszewski, M., & Fleury, C. (2001). An improved weighting method with multibounds formulation and convex programming for multicriteria structural optimization. *International Journal for Numerical Methods in Engineering, 52*(9), 889-902.

Zhang, Han, Z.-H., Li, W.-J., & Song, W.-P. (2008). Bilevel adaptive weighted sum method for multidisciplinary multi-objective optimization. *AIAA journal, 46*(10), 2611-2622.

Zhen, X., Wang, Y., Xu, S., Zhu, Y., Tao, C., Xu, T., & Song, M. (2012). The engine knock analysis–an overview. *Applied Energy, 92*, 628-636.

Zheng, Q., & Zhuang, D. (1995). Integral global minimization: algorithms, implementations and numerical tests. *Journal of Global Optimization, 7*(4), 421-454.

Zhigljavsky, A. A., & Pintér, J. (1991). *Theory of global random search*: Springer Netherlands.

Zhou, A., Qu, B.-Y., Li, H., Zhao, S.-Z., Suganthan, P. N., & Zhang, Q. (2011). Multiobjective evolutionary algorithms: A survey of the state of the art. *Swarm and Evolutionary Computation, 1*(1), 32-49.

Zitzler, E. (1999). *Evolutionary algorithms for multiobjective optimization: Methods and applications* (Vol. 63): Citeseer.

Zou, Liu, M., Kang, L., & He, J. (2004). *A high performance multi-objective evolutionary algorithm based on the principles of thermodynamics.* Paper presented at the Parallel Problem Solving from Nature-PPSN VIII.

Zou, Zhang, Q., Yang, J., Cloutier, A., Gragg, J., & Pena-Pitarch, E. (2011). *Determining weights of joint displacement objective function for standing reach tasks.* Paper presented at the First international symposium on digital human modeling, Lyon, France.

Zou, Zhang, Q., Yang, J. J., Boothby, R., Gragg, J., & Cloutier, A. (2011a). An alternative formulation for determining weights of joint displacement objective function in seated posture prediction *Digital Human Modeling* (pp. 231-242): Springer.

Zou, Zhang, Q., Yang, J. J., Cloutier, A., & Pena-Pitarch, E. (2012). Nonlinear inverse optimization approach for determining the weights of objective function in standing reach tasks. *Computers & Industrial Engineering, 63*(4), 791-801.

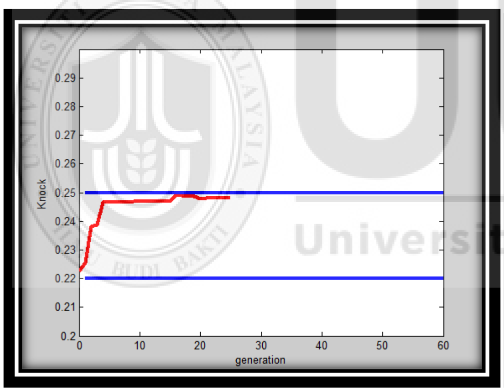
Universiti Utara Malaysia

APPENDIX A

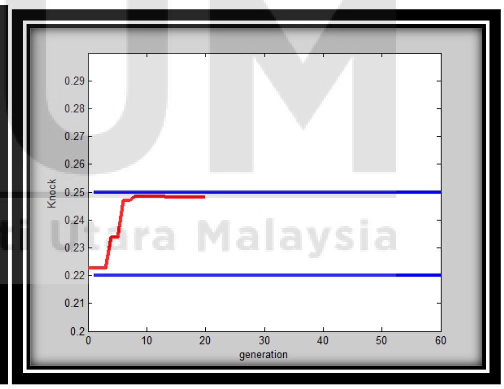
PROTON_TURBO CHARGE ENGINE

RUN	Elapsed time	Generations	Best Knock	Optimal Factors(Best solution)		
				TPS	RPM	TEMP.
1	0.030314	25	0.24803	80.013	1218.1	89.905
2	0.033244	20	0.24802	80.023	1199.9	90.135
3	0.042071	53	0.24799	80.021	1205.5	91.364
4	0.053894	60	0.24721	80.012	1215.4	91.382
5	0.041634	60	0.24836	80.345	1195.7	90.504
6	0.050171	36	0.24793	80.137	1066.5	89.494
7	0.026348	4	0.24792	80.013	1217.2	89.5
8	0.044953	60	0.24741	80.02	1202.7	90.839
9	0.036225	47	0.24793	80.148	1062.6	90.499
10	0.034941	25	0.24803	80.013	1218.1	89.905
11	0.032464	20	0.24802	80.023	1199.9	90.135

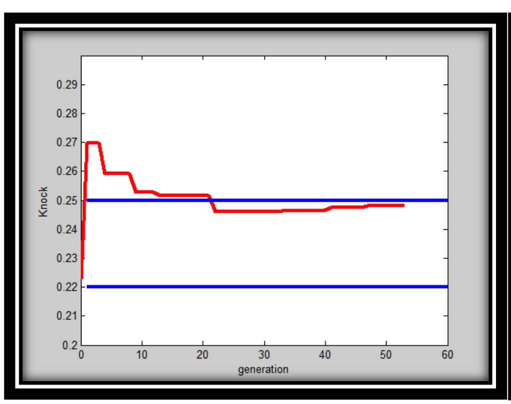
UUM
Universiti Utara Malaysia



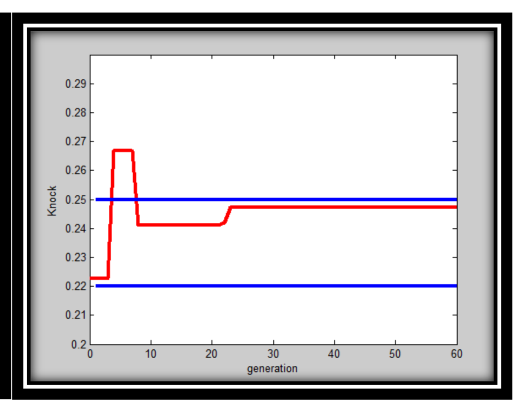
(1)



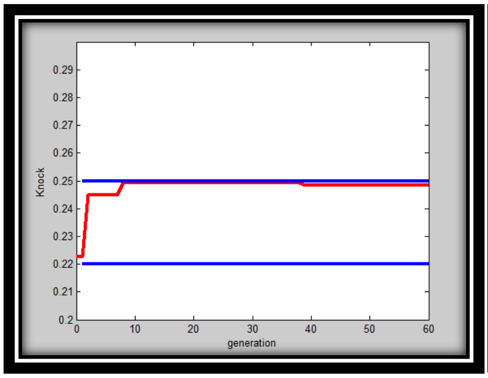
(2)



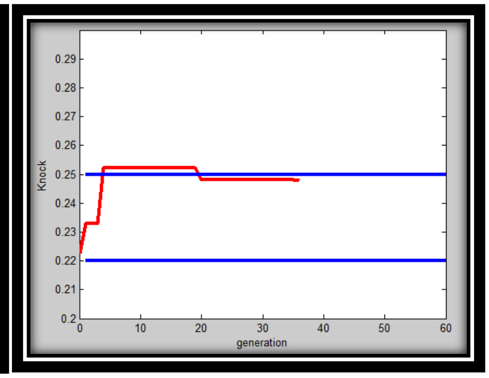
(2)



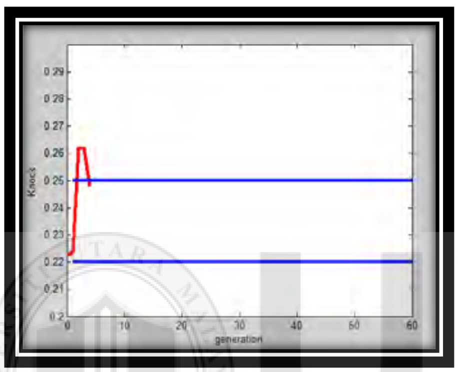
(4)



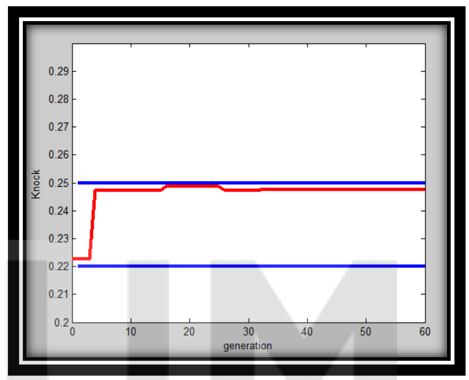
(5)



(6)

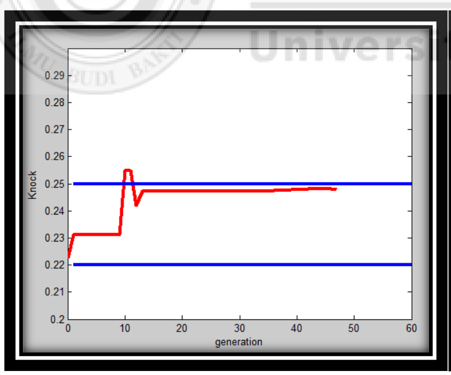


(7)

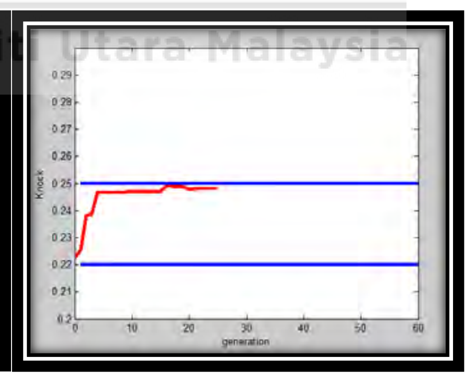


(8)

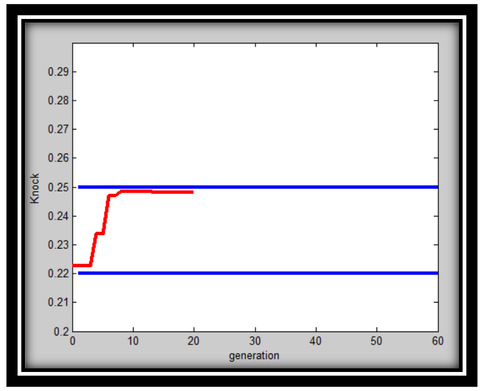
UUM
Universiti Utara Malaysia



(9)



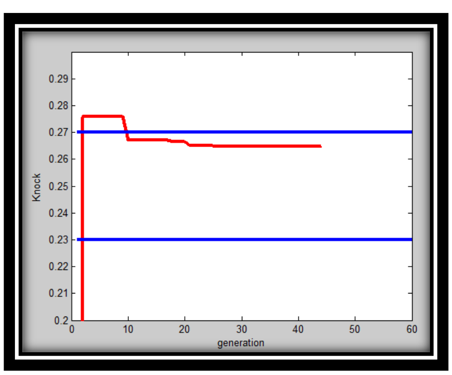
(10)



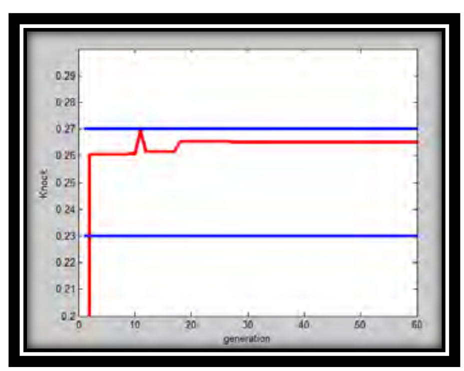
(11)

Appendix B
KIA_Motors_Sorento

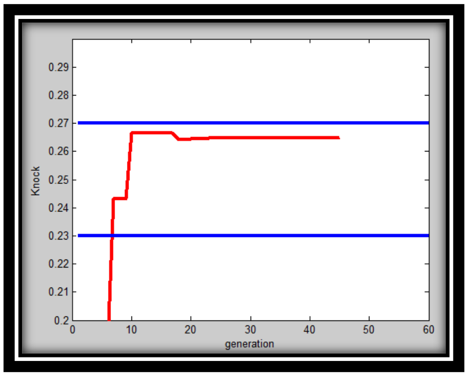
RUN	Elapsed time	Generations	Best Knock	Optimal Factors(Best solution)		
				TPS	RPM	TEMP.
1	0.046865	44	0.26499	7.6713	1717.5	91.582
2	0.054338	60	0.26514	7.562	2032.3	92.55
3	0.040090	45	0.26494	7.4681	2397.3	89.915
4	0.036770	43	0.26498	7.4644	2416.2	91.174
5	0.061880	54	0.26499	7.4904	2302.2	90.591
6	0.049794	57	0.265	7.4911	2296.3	87.882
7	0.045329	60	0.26469	7.6645	1729.9	90.926
8	0.044775	53	0.26498	7.6344	1814	95.045
9	0.054044	46	0.26503	7.7019	1658	94.302
10	0.063411	60	0.26464	7.6406	1784.4	87.068



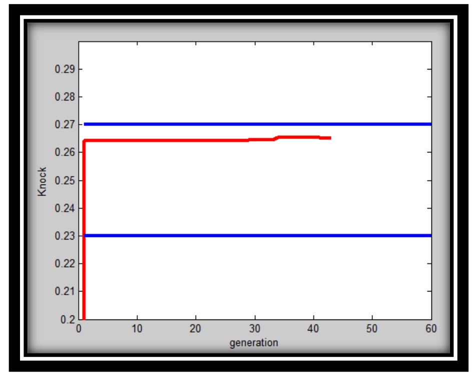
(1)



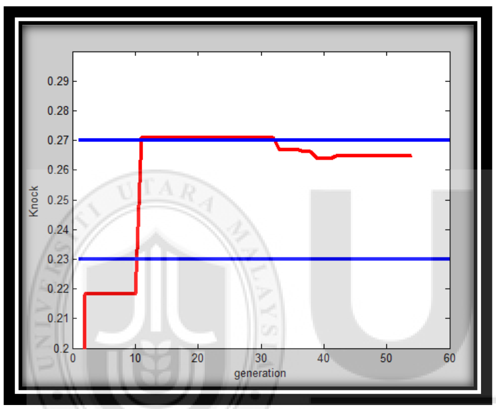
(2)



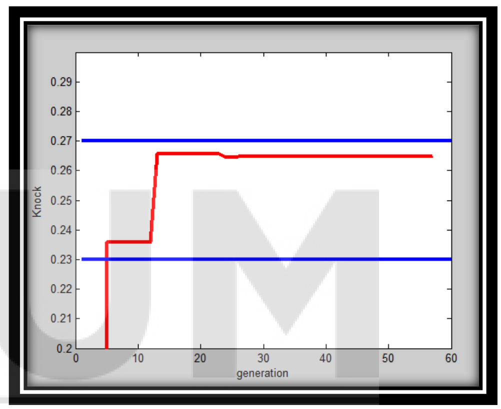
(3)



(4)

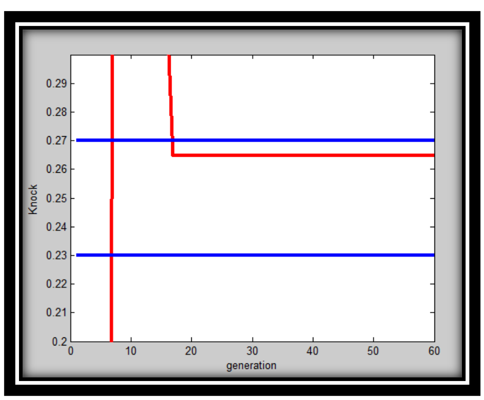


(5)

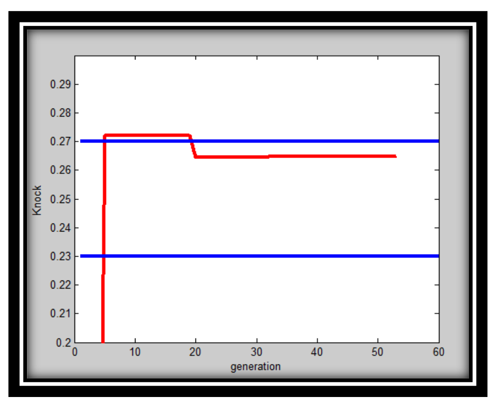


(6)

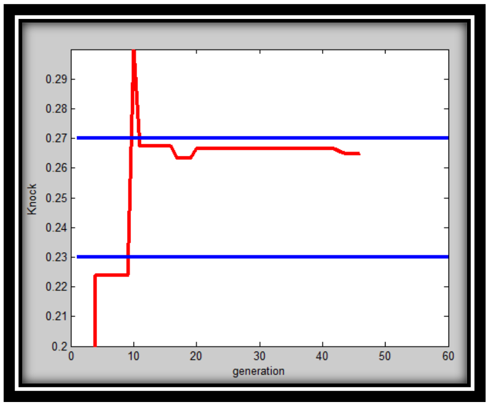
Universiti Utara Malaysia



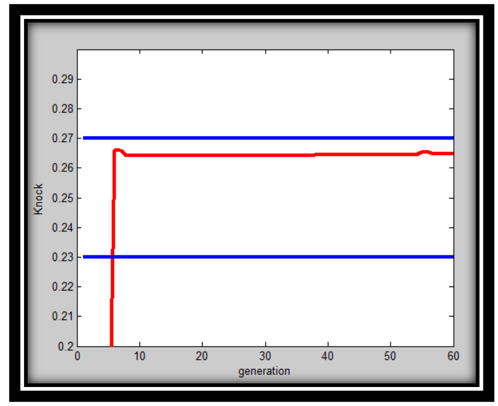
(7)



(8)



(9)

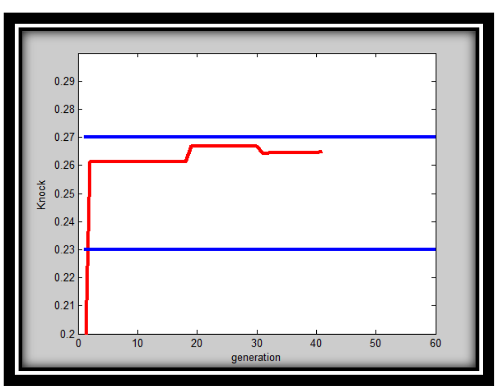


(10)

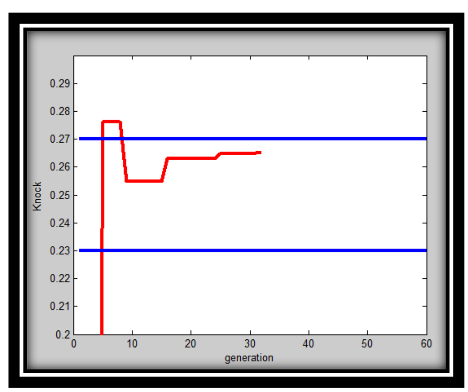
Appendix C
Hyundai_Genesis Engine

RUN	Elapsed time	Generations	Best Knock	Optimal Factors(Best solution)		
				TPS	RPM	TEMP.
1	0.041626	41	0.26504	7.6469	1776.1	91.387
2	0.032792	32	0.26498	7.6585	1753.9	95.858
3	0.046630	40	0.26494	7.5212	2180.6	92.85
4	0.043320	59	0.26507	7.4781	2355.7	90.784
5	0.028497	25	0.26498	7.4744	2372.4	91.715
6	0.041442	47	0.265	7.4675	2407	94.06
7	0.028881	25	0.26499	7.5968	1918.1	92.593
8	0.036631	35	0.26507	7.6751	1704.8	87.575
9	0.038024	27	0.26504	7.7449	1576.5	88.572
10	0.033767	31	0.26496	7.5119	2214.4	90.818

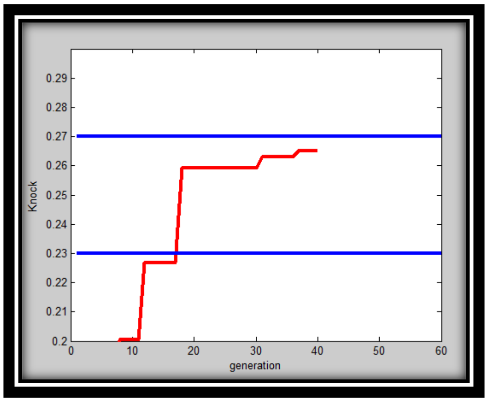
Universiti Utara Malaysia



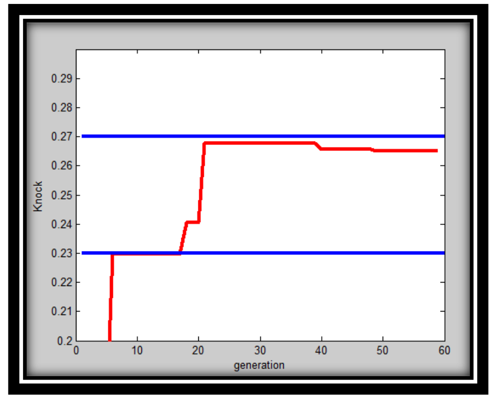
(1)



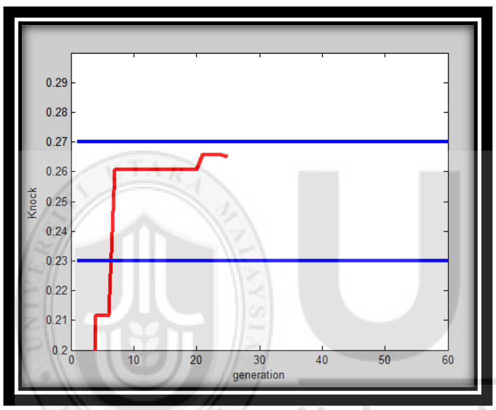
(2)



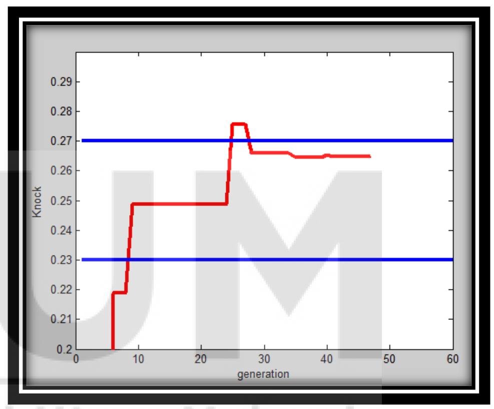
(3)



(4)

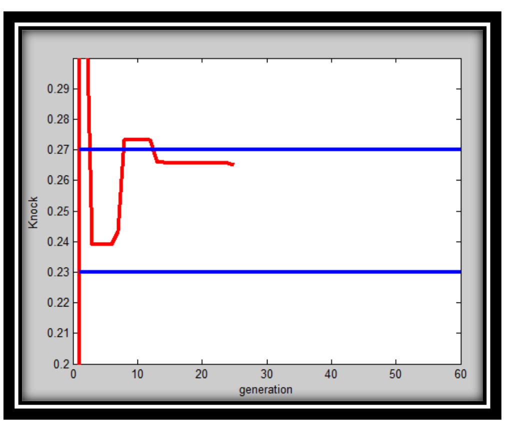


(5)

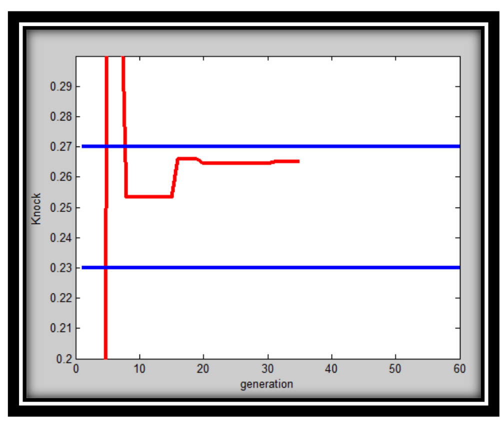


(6)

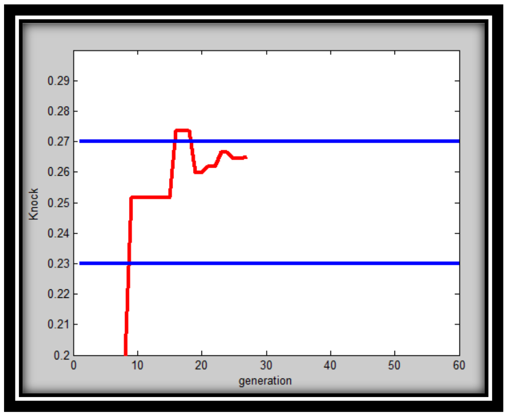
Universiti Utara Malaysia



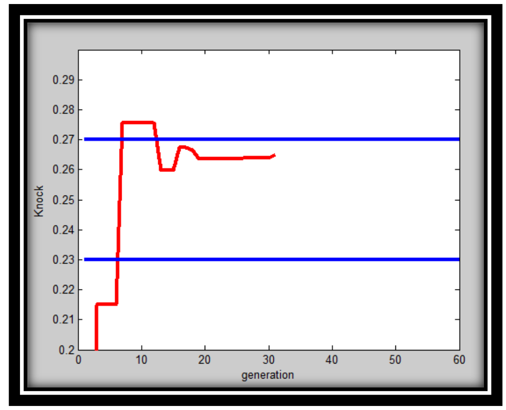
(7)



(8)



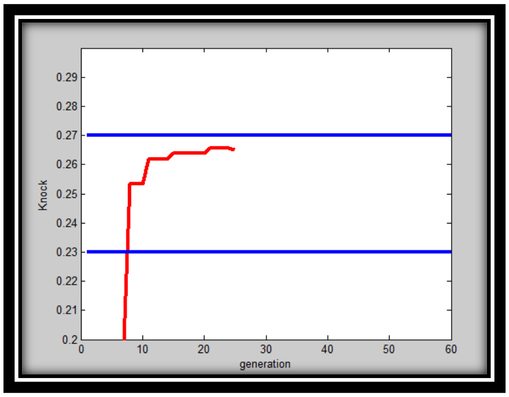
(9)



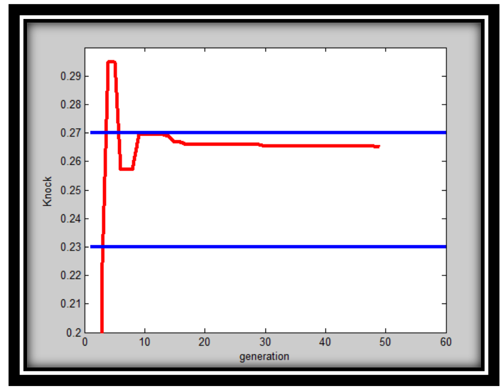
(10)

Appendix D
Dodeg Engine

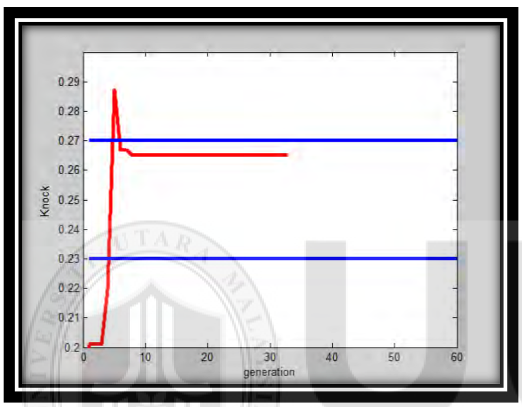
RUN	Elapsed time	Generations	Best Knock	Optimal Factors(Best solution)		
				TPS	RPM	TEMP.
1	0.036748	25	0.26496	7.8568	1482.4	53.379
2	0.050063	49	0.26494	7.5149	2199.8	60.493
3	0.035558	33	0.26507	7.5659	2012.9	58.467
4	0.049124	34	0.265	7.7616	1559.9	66.594
5	0.078471	60	0.26447	7.6748	1701.7	58.603
6	0.050132	49	0.26499	7.8438	1483.9	57.734
7	0.051138	27	0.26504	8.2077	2059.7	59.7
8	0.048178	19	0.26503	7.8392	1487.5	55.217
9	0.032778	19	0.26496	7.8403	1485.8	60.848
10	0.061364	60	0.26509	8.1502	1875.4	55



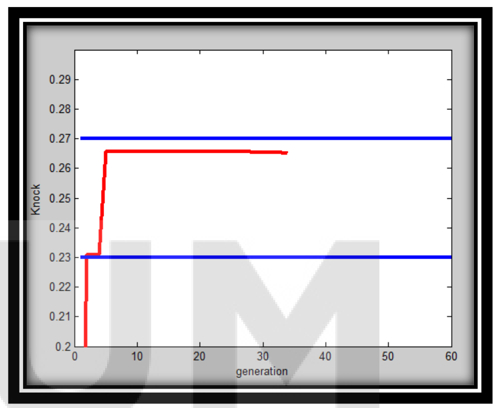
(1)



(2)

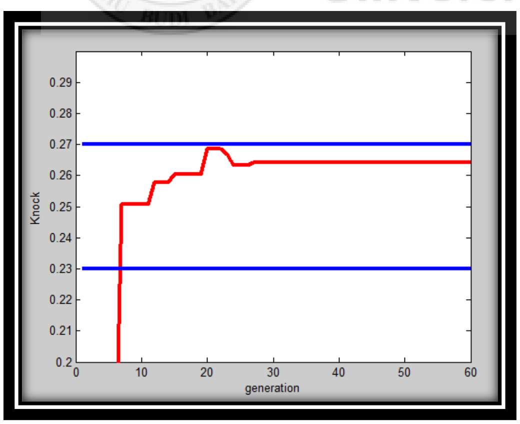


(3)

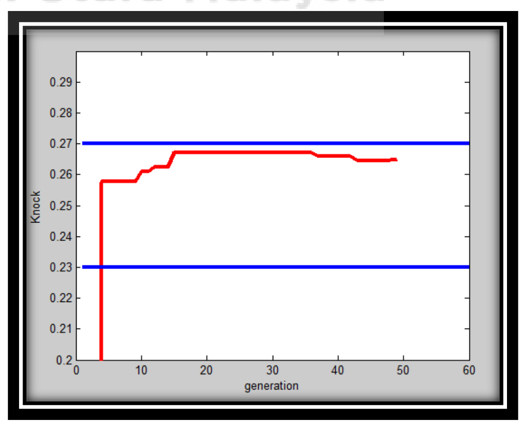


(4)

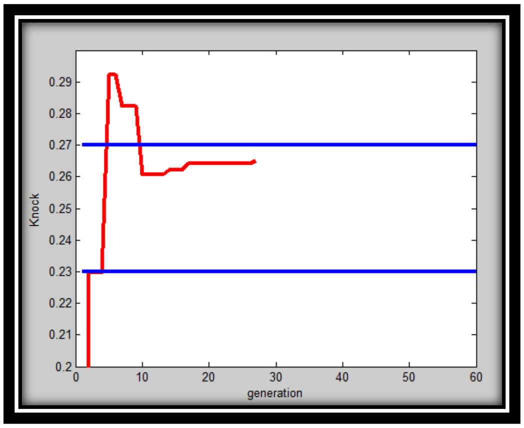
Universiti Utara Malaysia



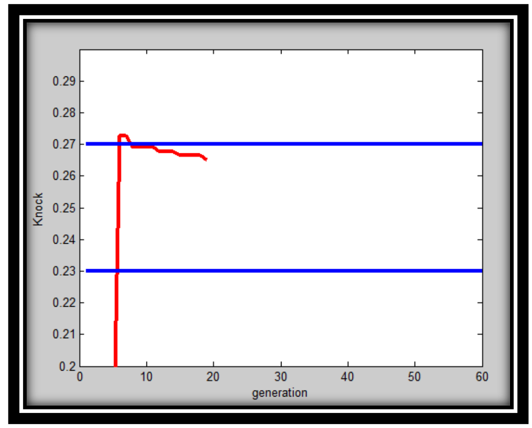
(5)



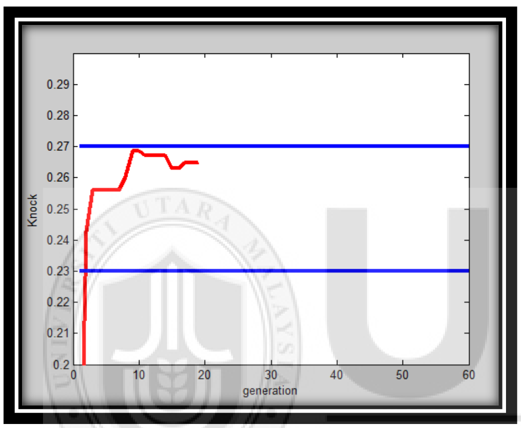
(6)



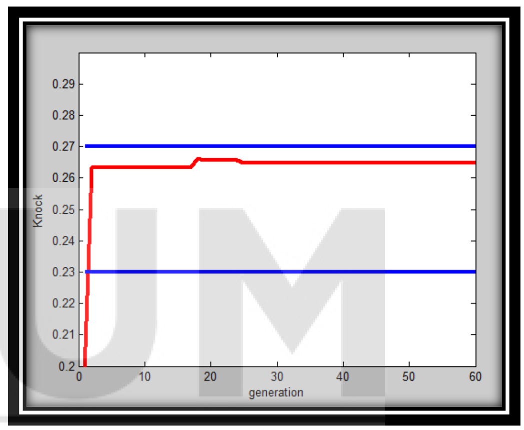
(7)



(8)



(9)



(10)

Universiti Utara Malaysia

APPENDIX E
DATA SETS

PROTON TURBO CHARGE

TPS	RPM	TEMP.	KNOCK
80.02320454129288	1000.5	90.1	0.27479553244339666
80.02182594762671	1000.4	89.9	0.3145224750739207
80.02204010507785	1000.3	90.1	0.3127932979287972
80.02101265030628	1000.5	90.3	0.32168359838959226
80.02027229444965	1000.4	89.3	0.34870855850050336
80.0210819446969	1000.4	90.3	0.34411675942900016
80.00625294512477	1000.5	90.6	0.292195760078057
80.00976142952834	1000.5	89.5	0.29028704089382235
80.01184026124425	1000.5	90.1	0.292558785175316
80.01235449856314	1000.4	89.9	0.31869671327074955
80.0123581456365	1000.4	89.8	0.31658149789903944
80.03112963132469	1999.9	90.0	0.17247724117107777
80.03151629223605	1999.8	90.3	0.17586395919059505
80.03051649428397	2000.0	90.2	0.1748822187824208
80.02979766144979	1999.8	90.2	0.18403744283795875
80.02973927908766	1999.9	89.8	0.18555168647629877
80.03167684373201	2000.0	90.1	0.1864885211388481
80.02031782539426	2999.8	90.3	0.40320785554277044
80.02346713056056	2999.9	90.1	0.39643188507843946
80.01904800901737	2999.9	89.5	0.4157507327499335
80.02087610673237	3000.0	90.3	0.47375244178007603
80.01998942460719	4001.0	90.0	0.6912158245182007
80.0198909043711	4001.0	89.8	0.7295861886218123
80.01836487517335	4001.0	90.2	0.7053926579266526
80.02004982298519	4999.6	90.4	0.7921056044720308
80.02011348712678	4999.5	90.1	0.8052391800473924
80.02067347250018	4999.5	89.9	0.8212442393102849
80.02113152956694	4999.5	89.9	0.8536828129098429

Universiti Utara Malaysia

DODEG

TPS	RPM	TEMP.	KNOCK
2.6	740.0	50.0	0.5
2.8	745.0	50.0	0.6
2.9	750.0	50.0	0.7
3.0	780.0	50.0	0.8
3.1	790.0	52.0	0.8
3.6	800.0	53.0	0.9
2.5	850.0	55.0	0.6
2.5	950.0	55.0	0.6
3.0	960.0	56.0	0.8
3.1	980.0	57.0	0.7
3.2	990.0	57.0	0.7
3.0	1000.0	59.0	0.7
3.1	1050.0	60.0	0.7
3.1	1100.0	60.0	0.7
3.2	1200.0	60.0	0.9
3.2	1300.0	60.0	0.9
3.1	1350.0	61.0	0.8
3.5	1400.0	61.0	1.4
3.5	1470.0	64.0	1.5
5.0	1900.0	66.0	1.6
6.4	2000.0	66.0	1.7
9.1	2200.0	67.0	1.9
10.0	2400.0	68.0	2.0
12.0	2600.0	68.0	2.1
2.6	740.0	50.0	0.5
2.8	745.0	50.0	0.6
2.9	750.0	50.0	0.7
3.0	780.0	50.0	0.8

Universiti Utara Malaysia

HYUNDAI-GENESIS

TPS	RPM	TEMP	KNOCK
0.4	541.0	87.0	-2.25
0.4	543.0	96.0	-2.25
0.4	546.0	90.0	-2.25
0.4	550.0	93.8	-2.25
0.4	552.0	88.5	-2.25
0.8	583.0	92.3	-2.25
0.8	588.0	93.3	-2.25
0.4	589.0	90.8	-2.25
0.4	595.0	91.5	-2.25
0.8	595.0	90.0	-2.25
0.8	629.0	90.8	-2.25
0.8	636.0	92.3	-2.25
1.2	699.0	91.5	-2.25
1.2	739.0	94.5	-2.25
0.8	742.0	91.5	-2.25
0.8	759.0	91.5	-2.25
1.6	856.0	90.8	-2.25
1.2	878.0	94.5	-2.25
1.2	893.0	93.0	-2.25
1.5	933.0	93.0	-2.25
1.2	950.0	91.5	-2.25
1.6	1027.0	93.0	-2.25
1.6	1043.0	93.0	-2.25
2.4	1321.0	91.5	-2.25
2.4	1329.0	95.3	-2.25
2.4	1341.0	96.0	-2.25
3.5	1646.0	93.8	-2.25
3.5	1688.0	93.8	-2.25
5.5	2316.0	92.3	-2.25
5.9	2514.0	93.0	-2.25
7.8	3156.0	92.3	-2.25
0.8	759.0	91.5	-2.25

Universiti Utara Malaysia

KIA MOTORS-SORENTO

TPS	RPM	TEMP.	KNOCK
1.3	665.0	92.0	0.0
1.38	670.0	91.0	0.0
1.44	675.0	90.8	0.0
1.74	689.0	89.3	0.0
1.74	695.0	89.7	0.0
1.74	705.0	90.0	0.0
1.79	720.0	90.0	0.0
1.9	736.0	90.0	0.0
2.2	745.0	90.0	0.0
2.5	850.0	90.2	0.0
2.9	940.0	90.4	0.0
3.5	1150.0	90.5	0.0
4.63	1576.0	90.8	0.0
4.6	1585.0	91.0	0.0
4.45	1630.0	92.0	0.0
4.35	1725.0	92.5	0.0
4.22	1896.0	93.0	0.0
4.22	1907.0	92.3	0.0
4.22	1915.0	91.5	0.0
4.28	1940.0	91.5	0.0
4.34	1975.0	92.0	0.0
4.5	1975.0	92.5	0.0
4.7	2035.0	92.0	0.0
5.0	2189.0	93.0	0.0
5.26	2240.0	94.0	0.0
5.87	2604.0	95.3	0.0
5.87	2616.0	94.5	0.0
6.4	2870.0	92.0	0.0
8.03	3417.0	90.0	0.0
8.13	3473.0	90.0	0.0

Universiti Utara Malaysia