# EVOLUTIONARY ALGORITHMS WITH AVERAGE CROSSOVER AND POWER HEURISTICS FOR AQUACULTURE DIET FORMULATION

**ROSSHAIRY ABD RAHMAN**

**DOCTOR OF PHILOSOPHY**
**UNIVERSITI UTARA MALAYSIA**
**2014**

# 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
Kedah Malaysia

# Abstrak

Industri penternakan akuakultur merupakan salah satu industri yang paling penting di Malaysia kerana ia menjana pendapatan kepada pertumbuhan ekonomi dan menghasilkan sumber makanan utama kepada negara. Salah satu tunggak dalam industry penternakan akuakultur merupakan formulasi diet makanan untuk haiwan, yang juga dikenali sebagai kombinasi atau formulasi bahan makanan.Walau bagaimanapun, kos operasi komponen pemakanan dalam industry akuakultura dalah yang paling mahal kosnya, dan ini menyebabkan banyak kajian dijalankan berkaitan formulasi diet. Kekurangan kajian yang melibatkan pembinaan model telah memberikan motivasi untuk mengkaji formulasi diet, iaitu mencari kombinasi terbaik daripada bahan makanan yang dapat memenuhi keperluan pemakanan dengan kos yang minimum. Oleh itu, tesis ini mengkaji penggunaan Algoritma Evolusi (EA) bagi mencadangkan penyelesaian formulasi diet untuk penternakan akuakultur, khususnya udang. Dalam usaha untuk mendapatkan kombinasi bahan yang terbaik, kaedah penapisan heuristik yang dikenali sebagai Heuristik Kuasa diperkenalkan di peringkat pemulaan dalam metodologi EA. Ia berupaya menapis beberapa bahan yang tidak diingini daripada senarai bahan pilihan yang telah dikenalpasti daripada pangkalan data, yang mana ia boleh membawa kepada satu penyelesaian yang tidak diingini. Kejayaan model EA yang dicadangkan ini juga bergantung kepada operator baharu bagi pemilihan dan penyilangan, yang dapat meningkatkan prestasi penyelesaian secara keseluruhan. Tiga model utama EA telah dibangunkandengan mekanisma pemulaan yang baharu, serta operator pemilihan dan operator penyilangan yang pelbagai. Keputusan kajian mendapati model EA-PH-RWS-Avg adalah yang paling berkesan dalam memberikan hasil penyelesaian terbaik dengan nilai penalti paling minimum. Model baharu yang dicadangkan ini adalah efisien dan mampu disesuaikan dengan perubahan dalam parameter, justeru dapat membantu pengguna menyelesaikan masalah berkaitan formulasi diet udang, khususnya menggunakan bahan tempatan. Selain itu, strategi formulasi diet ini juga menyediakan elemen berasaskan pilihan pengguna untuk menentu bahan pilihan makanan dan jumlah berat bahan yang sesuai.



**Kata Kunci**: Algoritma evolusi, Heuristik kuasa, Operator penyilangan purata, Formulasi diet, Kombinasi pemakanan

# Abstract

The aquaculture farming industry is one of the most important industries in Malaysia since it generates income to economic growth and produces main source of food for the nation. One of the pillars in aquaculture farming industries is formulation of food for the animal, which is also known as feed mix or diet formulation. However, the feed component in the aquaculture industry incurs the most expensive operational cost, and has drawn many studies regarding diet formulation. The lack of studies involving modelling approaches had motivated to embark on diet formulation, which searches for the best combination of feed ingredients while satisfying nutritional requirements at a minimum cost. Hence, this thesis investigates a potential approach of Evolutionary Algorithm (EA) to propose a diet formulation solution for aquaculture farming, specifically the shrimp. In order to obtain a good combination of ingredients in the feed, a filtering heuristics known as Power Heuristics was introduced in the initialization stage of the EA methodology. This methodology was capableof filtering certain unwanted ingredients which could lead to potential poor solutions. The success of the proposed EA also relies on a new selection and crossover operators that have improved the overall performance of the solutions. Hence, three main EA model variants were constructed with new initialization mechanism, diverse selection and crossover operators, whereby the proposed EA-PH-RWS-Avg Model emerged as the most effective in producing a good solution with the minimum penalty value. The newly proposed model is efficient and able to adapt to changes in the parameters, thus assists relevant users in managing the shrimp diet formulation issues, especially using local ingredients. Moreover, this diet formulation strategy provides user preference elements to choose from a range of preferred ingredients and the preferred total ingredient weights.

**Keywords**: Evolutionary algorithm, Power heuristics, Average crossover operator, Diet formulation, Feed mix

# Acknowledgement

In the name of Allah, the Beneficent, the Merciful.

This work simply could not have been possible without the assistance and encouragement from many others. First and foremost, I would like to thank the most important people that make this thesis possible. The person is my supervisor, Associate Professor Dr. Razamin Ramli. I sincerely thank her for her support and guidance throughout the journey of my studies. I would also like to thank my former supervisor the late Associate Professor Dr. Ang Chooi Leng for her motivation and inspiration during her supervision at the difficult early stage of my journey.

I would like to acknowledge the support from all the persons involved in the process of data gathering including research officers, manufacturers and farmers. I would also like to express my gratitude to Universiti Utara Malaysia for the sponsorship given. Thanks also to the Dean and management staff of the School of Quantitative Sciences.

I am also grateful to all my friends for being there with me throughout the ups and downs in my PhD journey. They include Adyda, Nerda, Shery, Linda, Kak Coknin, Kak Aza, Kak Zura, Kak Deqta, Kak Kiah, Ja, Haslinda, Huai Tein and many more whose names I simply cannot mention here simply because they are so many of them.

Finally, I am forever indebted to my family who has always been understanding and supportive throughout my study. My special thanks go to my beloved husband,Ku Hushaini Ku Aris, for his patience, care, love and prayer, and to my children, Nina, Alya, and Irham. Not to forget my special thanks to my mother, sisters, brothers, nephews and nieces for their constant support and prayer.Thank you all from the bottom of my heart. May Allah bless you all...

# Table of Contents

# List of Tables

# List of Figures

# List of Appendices

# List of Abbreviations

| | |
|---|---|
| ABC | Artificial Bee Colony |
| AI | Artificial Intelligence |
| BCGA | Binary Coded Genetic Algorithm |
| BSI | Bee Swarm Intelligence |
| CCP | Chance Constrained Programming |
| DOF | Department of Fisheries |
| DP | Dynamic Programming |
| DSS | Decision Support System |
| EA | Evolutionary Algorithm |
| EAA | Essential Amino Acids |
| FAO | Food and Agriculture Organization |
| FLP | Fuzzy Linear Programming |
| GA | Genetic Algorithm |
| GP | Goal Programming |
| GUI | Graphical User Interface |
| IFAH | International Federation for Animal Health |
| LP | Linear Programming |
| MCDM | Multi Criteria Decision Making |
| MCS | Monte Carlo Simulation |
| MOFP | Multi Objective Fractional Programming |
| MOP | Multi Objective Programming |
| MGP | Multi Goal Programming |
| NLP | Nonlinear Programming |
| NRC | National Research Council |
| OR | Operations Research |
| PSM | Pearson's square method |
| QP | Quadratic Programming |
| RCGA | Real Coded Genetic Algorithm |
| RPV | Roulette penalty value |
| RF | Risk Formulation |
| RT | Roulette-Tournament |

| | |
|---|---|
| RWS | Roulette Wheel Selection |
| SAE | Simultaneous Algebraic Equations |
| Sec | Second |
| TE | Trial and Error method |
| QB | Queen-Bee |

# CHAPTER 1

# INTRODUCTION

Animal source food is important for humans to avoid malnutrition since it provides a lot of nutrients needed by a human body often limited in a diet (Demment, Youngy, & Sensenig, 2003; Neumann, Harris, & Rogers, 2002). These nutrients include protein, iron, vitamin, carbohydrate, potassium and sodium, which contribute in generating new tissues and producing energy, and have diversified benefits to humans. However, only healthy aquaculture can provide healthy food in adequate quantity for human consumption (Hansard Team Kenya National Assembly, 1993; International Federation for Animal Health [IFAH], 2011). Among the animals that contribute a good source of food are fish, mussels and shrimps.

## 1.1 Challenges in aquaculture industry

Growth in the world population has increased demand for healthy animal source food including aquaculture produce. As captured aquacultures can no longer meet the high market demand, the farming industry is forced to increase the production of farmed animals to fulfil the current needs. Like other industries, the main objective of a food producing industry is to generate maximum income and profit. Therefore, farmers have to strategize in order to minimize their production costs and sell their produce at the highest possible price. In addition to market demand and size, farmed animal price depends on its appearance such as stress and unhealthy eyes (Blue et al., 2007). Sufficient nutritional need is important to obtain good appearance and healthy body, thus contributing to a higher sale market value. Farmers need to provide enough nutritious food to ensure that their farmed animals receive adequate nutrition.

Over nutrition or under nutrition will lead to several problems such as diseases, viral infection and stunted growth, which might seriously affect the animals' health. If such crisis continues, the animals will die; hence farmers have to bear substantial losses.

In an aquaculture industry, operational costs include feeding cost, staff salary, maintenance cost, and utilities. However, of all these costs, the feeding cost contributes to more than half of the total operating costs (Abu Hassan, Hanafi, Che Musa, & Pathmasothy, 1988; Becker, 2008; Chappell, 1974). Therefore, this industry has to develop a strategy to reduce the feed cost without compromising the animal's health. The process of getting reliable feed with low cost starts with the ingredient selection until the pallet processing phase. One of the vital processes is the formulation of the ingredients to satisfy nutritional needs (Tacon, 1990).

Globally, shrimp is an important aquaculture produce after fish. In Asian countries including Malaysia the highest aquaculture production is shrimp (Food and Agriculture Organization of the United Nations [FAO, 2013]), and it contributes to the economic growth of the country. Therefore, a shrimp farming industry is explored in an attempt to increase the contribution of this industry to the country.

## 1.2 Farmed shrimp

The world's production of farmed shrimp was 3,930,059 tonnes in 2011 (FAO, 2013) in which 85.9% of farmed shrimp production comes from Asia, with China, Thailand, Vietnam, and Indonesia being the top four producers (FAO, 2013). Figure 1.1 shows the top ten leading countries in the world in farmed shrimp production. China produces almost 40% of the world farmed shrimp production followed by

Thailand, Vietnam, and Indonesia with values of 13%, 13%, and 10%, respectively. The American continent, on the other hand, contributes to 13.8% of the world's production of farmed shrimp, with the three major producers (Ecuador, Mexico and Brazil) accounting for 11%. Malaysia is the ninth world producer producing 67505 tonnes of farmed shrimp for the year 2011, which contributed to 2% of total world production. The remaining 7% comes from other countries that produce less than 2% of the world's farmed shrimp production.

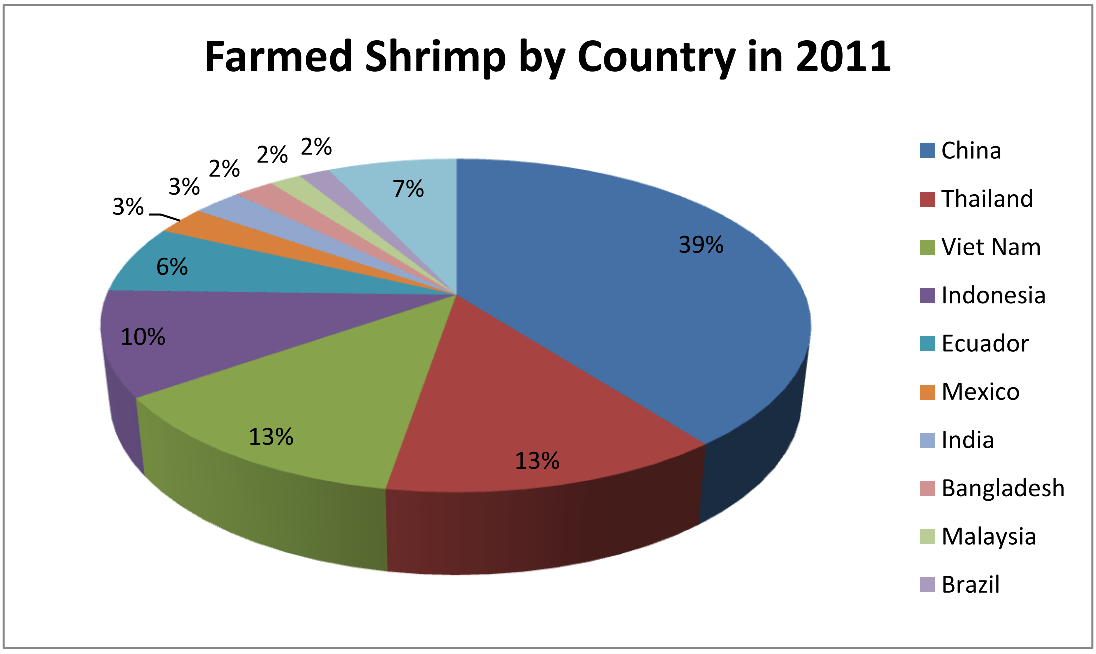



*Figure 1.1*. Farmed Shrimp Production by Country (FAO, 2013)

### 1.3 Operational cost of shrimp farming

Operational cost is a big issue in a shrimp farming industry. Farmers incur feed cost, labour cost, material, seed and utilities such as electricity and water consumption. Feed cost contributes the highest portion of the operational cost (Hertrampf, Biswal, & Mishra, 2005; Shang, Leung & Ling, 1998). High feed cost is therefore a major hurdle to majority of shrimp farmers. Feed cost is crucial to be researched on in order to provide sustainability of the shrimp industry. Therefore, there is a need to focus on shrimp feed to cut down feeding cost and help farmers earn more profit as well.

### 1.4 Challenges in shrimp feed

Based on observations and interviews with experts in Department of Fisheries, Malaysia, globally, most of the shrimp feed production is dominated by China, Thailand, and Indonesia. The Malaysian shrimp diet or feed production cannot survive because of the competition in the feed industry due to the relatively good quality of the imported feed. Most shrimp farmers in Malaysia use imported shrimp feed as the price is lower but this exercise has posed a huge loss to the nation. The lower price of the imported feed is due to the low cost of ingredient in the dominated countries. But the quality of shrimp feed highly depends on two factors i.e. nutrients and ingredients which are discussed as follows.

#### 1.4.1 Nutrients

Shrimp nutrient is a complex subject because the nutritional requirements of shrimp change with each stage of life cycle (i.e., larval, nursery, juvenile, adult). Thus, shrimp feeds must be specially formulated for different states of life (Fox, Treece, & Sanchez, 2001). Juvenile shrimp requires higher nutrition value especially protein than shrimps in other states of life. For this reason, most studies have focused on juvenile shrimp (Forster, & Decamp, 2002; Fox, Lawrence, & Li-Chan, 1995; Roy, Davis, Saoud, & Henry, 2007; Tacon, Cody, Conquest, Divakaran, Pascual, Zenteno, Cuzon, Suárez, Sánchez, & Gaxiola, 2004; Venero, Davis, & Rouse, 2007).

Shrimp requires specific nutritional requirements such as protein, lipid, ash and fibre for its growth. Protein is the most expensive nutrient sources that contains two different types of protein known as crude protein and amino acid. There are 22 amino acids that are commonly found in proteins (Lazo & Davis, 2000), but shrimp requires ten essential amino acids (EAA) in specific value to achieve its optimal growth. Unfortunately, only a small number of studies reported on amino acid (Fox, Davis, Wilson & Lawrence, 2006) due to lack of information on the requirements for total amino acids. Most commercial formulations using moderate levels of protein do not include minimum restrictions for amino acids in their formulation (Fox et al., 2006). This problem also happens in Malaysia when several approximate nutrients are taken into account (protein, fat, ash, fibre, calcium and phosphorus) without considering amino acids.

### 1.4.2 Ingredient selection

Most aquacultures including shrimp are very rigid towards their diet. Shrimp might not eat if a pallet provided is not palatable enough or if the colour and smell are not attractive enough. Therefore, ingredient restriction is required to ensure the shrimp diet contains sufficient nutrient with appropriate characteristics to attract it. Fish meal is one of the most expensive ingredients in a shrimp diet but also a primary source of protein because of its known nutritional and palatability characteristics (Cruz-Suárez, Nieto-L, Guajardo-Barbosa, Tapia-Salazar, Scholz, & Ricque-Marie, 2007). For this reason, it is good if fish meal can be included in the shrimp diet to acquire a more palatable and nutritious diet.

Instead of fish meal, a shrimp diet may consist of a combination of ingredients to generate high nutrient value. The substitution of imported ingredient with local available ingredient is one of the solutions to trim down the feed cost. Therefore, a computerized shrimp diet formulation is an advantage as it allows users to choose local ingredients or lower cost ingredients based on market price with ease.

**1.5 Issues in diet formulation approaches**

A diet formulation is a list of types and amount of ingredients to be included in an animal diet (Render et al., 2006). Currently, a great number of publications related to shrimp diet is available (see Deng, Mai, Ai, Zhang, Wang, Xu et al., 2006; Xie, Tian, Jin, Yang, Liang & Liu, 2014; Oujifard, Seyfabadi, Kenari & Rezaei, 2012), but mostly focused on experimental designs (see Bauer, Prentice-Hernandez, Tesser, Wasielesky Jr., Poersch, 2012; Ju, Deng, Dominy, 2012). Little attention was given to modelling approaches or formulation of shrimp diet, even though the formulation part is vital as using different models will present different results.

Not limited that of shrimp, diet formulation problems also encompass fish, cattle, poultry and cow, among others. Researchers attempted to formulate these animal diets used several approaches to improve the quality of feeds. These approaches are algebraic type (Afolayan & Afolayan, 2008; Gillespie & Flanders, 2009), optimization (Udo, Ndome, & Asuquo, 2011a; Saxena, 2011), heuristics (Pathumnakul, Ittiphalin, Piewthongngam, & Rujikietkumjorn, 2011) and integrated approaches (Thammanivit & Charnsethikul, 2013). An algebraic approach was employed for simple problems only as it cannot cater for problems with more than two nutrients (Gillespie & Flanders, 2009). Meanwhile, optimization is good and widely applied in diet formulation for animal. Unfortunately, the optimization approaches cannot be used to solve complex problems in polynomial time (Burke & Kendall, 2005). An integrated approach is a combination of two or more methods, which optimization and heuristics are commonly integrated with any other methods for a diet formulation domain.

A heuristics approach is relatively new to the diet formulation problem as compared to optimization since only a few studies have used it. This approach is able to find a near optimal solution for a complex problem in a reasonable time (Burke & Kendall, 2005). Based on discussions in Section 2.3.3.3, the heuristics approach is the best technique to solve a diet formulation problem (e.g. see Pathumnakul et al., 2011). It consists of metaheuristics which is known as a higher level of heuristics (Talbi, 2009). One particular example is the Evolutionary Algorithm (EA) which belongs to the family of metaheuristics that is population-based in nature.

An EA starts with initialization, and then follows by selection, crossover and mutation. These processes continue in many generations until a termination criterion is reached. Initialization is based on random and semi-random operations while selection consists of Roulette Wheel, Tournament and Ranking operations or sub-heuristics. The commonly used crossover are One-Point, Multi-Point and Uniform types. Meanwhile, a mutation operator is often designed based on type of encoding such as bits and real number. An allele in a chromosome depicts the type of representation in either by bits or real valued. In fact, a mutation operator is also designed based on constrained or unconstrained problem such as the Power Mutation by Deep and Thakur (2007).

EAs have shown good solutions in solving a real value representation problem such as the diet formulation problem. The first research using the EA in diet formulation was employed by Furuya, Satake, and Minami (1997) which studied on general diet formulation. The effort was followed by Şahman, Çunkaş, İnal, İnal, Coşkun, and Taşkiran (2009) that focused on diet formulation for poultry and cattle. Şahman et al. (2009) used the semi-random initialization, Tournament selection and enhanced One-Point Crossover adopted from Randly and Sue (1998). The mutation operator is similar to crossover that is based on probability value generated randomly. These limited studies that explained EAs in diet formulation problems have laid out a conceptualization and thus, motivates us to further explore the EA and diet formulation problem.

### 1.6 Problem statement

Shrimp is the most important aquaculture commodity in Malaysia that contributed to the economic growth (FAO, 2013). In shrimp industry, shrimp feed was identified as the most expensive component of the total operation cost as highlighted by Hertrampf et al. (2005) and Shang et al. (1998). Therefore, providing a better shrimp feed in a minimum cost is very valuable to farmers to help them increase their profits, which this issue is the focus of our research.

Through discussions in previous sections, it is known that shrimp feeds comprise of ingredients with specific nutritional requirements such as protein, carbohydrate, lipid, vitamin and mineral for shrimp growth. From the industry perspective, these combinations of nutrients are very expensive since they mostly come from fish meal, which is one of the important and most expensive ingredients identified by Cruz-Suárez et al. (2007). Due to that, much work and studies were inclined towards shrimp feed formulation, but they mostly used experimental design approaches to find the best combination of ingredients. However, it has been identified that limited work has been done in the alternative approach of modelling or formulation of the ingredients for shrimp diet. Hence, our work focuses on the feed or diet formulation of the ingredients to obtain specific nutrient requirements and intends to fill the knowledge gap in this area.

As highlighted by Burke and Kendall (2005), heuristic approaches including the EA are able to find good near optimal solutions for complex combinatorial problems in reasonable times. However, to our knowledge, there is no literature that has attempted to formulate ingredients for the shrimp diet in a heuristic manner. The most similar works that focused on diet formulation are by Furuya et al. (1997) and Şahman et al. (2009). But Furuya et al. (1997) studied on general diet, while Şahman et al. (2009) studied on poultry and cattle diet formulations. In this sense, our work is different from theirs as we focus on shrimp diet formulation with complexity of many nutrients and ingredients that require specific restrictions to be fulfilled. This new shrimp diet formulation study is further enriched by the employment of the recommended EA.

As is known, an EA has multi-stage operations which are initialization, selection of parents, crossover and mutation. Apart from the random type of initialization, the semi-random type of initialization has also proven to be effective such has been done by Şahman et al. (2009). Shrimp diet formulation involves many ingredients with strict nutrient requirements. Hence, a semi-random initialization operation is most suitable since it possess the filtering ability, which is in line with the objective of achieving nutrient strictness. Therefore, our work takes into consideration this semi-random initialization and further improves with a new alternative initialization operators based on the work by Deep and Thakur (2007). Our improvement differs from Deep and Thakur (2007) in the computation of obtaining new ingredient values in the EA chromosomes.

Another important operator in an EA is the crossover, where Şahman et al. (2009) used standard crossover operators like the One-Point and Multiple-Points. Since our shrimp ingredients are represented by real-valued alleles in the chromosome of the EA, an alternative crossover operator similar to the One-Point crossover is deemed necessary for the performance of the whole EA process. Subsequently, we explore on the possibility of a new alternative crossover operator suitable for real-valued chromosomes.

## 1.7 Research question

1. What is the important nutrient for shrimp diet?
2. What is the best technique to formulate shrimp diet?
3. What operators can be improved in EA?
4. How efficient is the technique?

## 1.8 Research objectives

The primary objective of this thesis is to develop a model that can lead to the creation of shrimp feed mix that will be able to meet the nutritional requirements for effective production. In order to reach the goal, some specific objectives need to be fulfilled. These are as follows:

1. To identify the maximum and minimum requirements of shrimp feed in various aspects.

2. To construct a new filtering heuristics known as Power Heuristics as part of the initialization procedure that is capable of filtering some combinations of ingredients from a selected database of choices, which could lead to potentially poor solution.
3. To construct a new crossover operator known as Average Crossover that is able to produce a potentially good solution.
4. To conduct a comparative evaluation on the solutions based on several evolutionary models generated and what-if analyses.

## 1.9 Research contribution

1. The main contribution of this research is the development of Evolutionary Algorithm (EA) model for the aquaculture diet formulation. In this research several new operators or procedures have been proposed in three stages of EA: initialization, selection, and crossover. The package of the methodology design is able to produce efficient diet for shrimp.
2. In order to suit the diet formulation problem into the research framework, special heuristic technique known as *Power Heuristics* is developed for filtering purposes in the initialization stage, in order to filter some combinations of ingredients from a selected database of choices, which to lead to potentially poor solution. The inclusion of this technique will allow more ingredients to be included in the formulation, and thus choosing the most suitable combination of ingredients.
3. A new crossover operator called *Average Crossover* is introduced. This operator is able to obtain better acceptable best-so-far solutions. The comparison with standard One-Point crossover is presented in Chapter Six.

4. As a side contribution, two existing selection techniques, Roulette Wheel Selection and Binary Tournament Selection, are combined together to produce a new selection procedure named *Roulette-Tournament*.
5. This study is the first attempt in Malaysia that concentrates on feed formulation problem using a mathematical approach. In addition to low cost, this proposed approach considers several aspects such as balanced in nutrients, ration characteristics, user preference of ingredients, and preference of bulk size. There are 16 nutrients with 91 ingredients in the database. There are also two nutrient combinations and one nutrient ratio is taken into consideration.
6. Bulk size based on user preference is introduced in this study in order to avoid waste as the bulk size can be flexibly designed and not necessarily set at 100 kg. Thus, the user has the opportunity to design the appropriate budget depends on his/her needs.
7. The proposed model can be used by the department of fisheries (DOF) Malaysia. Besides juvenile Whiteleg shrimp, other aquacultures can use the model with some adjustment on the nutritional requirements value of specific species. Other than DOF, nutritionists, manufacturers and farmers may also use this prototype to obtain an effective solution at a low cost.

### 1.10 Scope of the study

In order to determine the nutrient composition in each ingredient, this research uses the feed ingredient available in two reports published by the National Research Council (NRC) in 1982 and 2011. These reports are recommended by two experts in aquaculture nutrition. They are from The National Prawn Fry Production and

Research Centre (NAPFRE) in Pulau Sayak, Kedah, and from Universiti Putra Malaysia (UPM) in Serdang, Selangor.

This research focuses on the requirement of nutrients for juvenile Whiteleg shrimp. Nutrients needed in this research are based on past studies (Akiyama, 1992; NRC, 2011) and expert recommendation.

A nutrient value in each ingredient changes over time. In fact, the price of the ingredients also varies when the ingredients are bought in large quantities. However, in this research we do not consider this factor.

Evaluation of the proposed diet formulation is done by comparing it with the nutritional requirements of Whiteleg shrimp. No testing on experimental design in the pond was done due to lack of sources and time limitation.

### 1.11 Definition related to animal diet formulation problem

The following are terms and the definitions that are specific to the field of diet, in particular the animal diet that will be used in this thesis.

#### 1.11.1 Amino acid

Amino acid is the building block of protein which is divided into essential and non-essential amino acids. Non-essential or dispensable amino acids are synthesized in the body, while essential amino acids refer to nutrients that are not synthesized within the body (Lazo & Davis, 2000). In this research, essential amino acids (EAA) are investigated. The amino acids that fall within this category include lysine, leucine, arginine, threonine, methionine, tryptophan, histidine, phenylalanine, isoleucine, and valine.

### 1.11.2 Crude protein

Crude protein can be defined as an estimated amount of the total protein content of a feed determined by analysing the nitrogen content of the feed. The result is then multiplied by 6.25 (Ministry of Agriculture, Food and Rural Affairs, 2012). Crude protein consists of true protein and other nitrogen containing substances such as ammonia, amino acids and nitrates.

### 1.11.3 Nutrient

"Nutrient is a chemical or food that provides what is needed for plants or animals to live and grow" (Longman, 2003). Many nutrients should be taken into consideration in formulating a shrimp diet such as crude protein, ash, fibre, lipid, calcium, phosphorus and several types of amino acids.

### 1.11.4 Ration

Ration is a fixed allowance of total feed for an animal per day (Longman, 2003). It usually specifies the individual ingredients and the amounts of the specific nutrients such as carbohydrate, fibre, minerals and vitamins.

## 1.12 Thesis outline

This chapter has described the background of the problem and highlighted the significance of this research. This chapter has also discussed the contribution of this study towards the body of knowledge and diet formulation problem domain.

Chapter Two presents an overview of animal diet formulation problem. It draws attention to the list of techniques that has been used in this problem to solve the issues that have arisen in diet problem beginning from the year 1939, where the history of this problem started. The gap in animal diet formulation problem is identified from previous researches addressed in Chapter Two.

The foundation of the proposed methodology is observed in Chapter Three where it examines several procedures in previous work. This chapter highlights the concept of Evolutionary Algorithms by introducing the taxonomy of heuristics. Consequently, these will be the base for the contributions in this research.

Chapter Four explains the designed research methodology to achieve the objectives as described in this chapter. The first step in the methodology is problem definition. This is then followed by data collection, model formulation, model development, model validation and model verification.

Chapter Five describes the development of a diet formulation prototype. A detailed procedure of each step in the proposed algorithm is discussed in this chapter. An explanation on how the prototype is generated is discussed along with the illustration of prototype interface. Then, the implementation of the algorithm is tested using real data of juvenile shrimp in Chapter Six. Several experimentations are done for testing purposes to ensure the proposed algorithm is comparable with other established procedures. The results obtained from the experimentation are discussed.

Finally, in Chapter Seven a conclusion of the thesis is presented. A summary of the proposed methodology and the successful accomplishment of the research objective are described. Some recommendations on future work are proposed in this chapter.

# CHAPTER 2

# REVIEW OF DIET FORMULATION PROBLEM

Based on the discussion in Chapter 1, an investigation of a shrimp diet formulation is crucial. Therefore, this chapter begins by introducing the diet problem. Then, the factors that influence the diet formulation which becomes the objective function and constraint are discussed. After that, techniques and approaches that have been previously applied in the field of diet formulation are reviewed. They are algebraic, optimization, search, and integrated approaches. Next, the development of a computerized system in the diet formulation industry is offered. This chapter ends with a summary.

## 2.1 Introduction to diet problem

A diet problem occurs in humans and farmed animals. In general, a diet problem for humans and farmed animals is similar in its basic formulation (Eckstein, 1970). However, the term "menu planning problem" refers to the methods used for planning menus for the human diet (Eckstein, 1970), since it can also be defined as the scheduling of meals associated with the person's needs during the time horizon (Balintfy, Ross, Sinha, & Zoltners, 1978). For human, a variety of menu is a must to avoid dullness of the same menu. Consequently, the formulation of menu planning for human must include variety as one of the factors along with other important factors such as nutritional value and price. The diet problem for human has been widely studied in hospital patients, nursery (Mohd Noor, Abdul Rahman, & Sulaiman, 2007), athletes (Alkhazaleh, 2010), and boarding school (Mohd Razali, 2011).

In agricultural applications, especially for farmed animals, the diet problem is known as the feed mix problem (Render, Stair, & Hanna, 2006). In the field of dairy science, this problem is also known as ration formulation (Lara, 1993; Al-Deseit, 2009), diet formulation (Munford, 1996; Polimeno, Rehman, Neal, & Yates, 1999), or feed formulation (Afolayan & Afolayan, 2008; Villamide et al., 2009). In this research, ‘diet formulation’, ‘feed formulation’ and ‘feed mix’, are used interchangeably. The diet formulation problem in farmed animals differs from a menu planning problem as the former does not involve a variety of menu. But instead it involves a combination of several feed ingredients in specific quantities in order to satisfy nutritional restrictions at a minimum cost (Render et al., 2006). It can be concluded that diet formulation is a problem to find the best quantities of each appropriate ingredient that are able to fulfil decision-making criteria at a low cost.

The diet problem was first investigated by George Stigler in 1939 (Dantzig, 1990). Stigler recommended a diet problem for a moderately active man weighing around 154 pounds. His diet problem solved 77 types of food available with nine nutrients, which yielded a diet consisting of wheat flour, evaporated milk, cabbage, spinach, dried navy beans, pancake flour, and pork liver (Dantzig, 1990). Stigler used a Trial and Error technique to solve his diet problem and obtained the diet that cost only$39.93 per year. This cost was considered good enough at that time. However, Stigler’s diet problem could not satisfy many people since the types of food recommended contained less palatability. In addition, Stigler expected that this same diet would be eaten every day.

Later, in 1947 Jack Laderman introduced a simplex method to measure the least cost diet proposed by Stigler (Dantzig, 1990). This calculation became the first large scale computational problem in history. It took nine clerks using hand calculators in 120 days to produce the optimal solution of $39.69 per year. The result showed that the Stigler's diet which was solved by a trial and error technique was off by only 24 cents per year. Stigler's diet was then reformulated by George Dantzig known as linear programming (LP) model (Dantzig, 1990). After that, many researchers have expanded Stigler's diet to menu planning with the intention to produce palatable and diverse nutritious diet choices (Babu & Sanyal, 2009).

The history of diet formulation problem in animals began in 1951 when Frederick V. Waugh, who was from a council of economic adviser, tried to apply the same technique Stigler used for human. Waugh found that his research was beneficial in helping nutritionists design an animal diet. Since then, a lot of researchers have enhanced the diet formulation problems by adding more variables and solving them using various techniques. The variables or factors that contribute towards diet formulation are discussed in the next section.

## 2.2 Factors influencing diet formulation

In modelling a shrimp diet, the complexity of the problem increases with the involvement of many decision criteria. These criteria include balanced in nutrients, bulkiness of ration, ration characteristics, variability, user preference of ingredients, and environmental factor. Minimizing the feed cost is the main objective to achieve, and the constraints can be any of these decision criteria or some combinations of them.

### 2.2.1 Balanced in nutrient

Balanced in nutrient refers to the quantity of nutrients obtained in the final ration is sufficient based on the animal characteristics such as type and age. This factor includes several aspects, which are the restriction of nutrient (lower bound and upper bound) and nutrient ratios (Glen, 1980; Munford, 1989b). In fact, the type of suitable ingredients varies depending on the digestibility ability of the animal. Balanced in nutrient is the main concern in a diet problem and is considered by many researchers. Nutrient ratio is also considered by many previous researchers (Alexander, Morel & Wood, 2006; Furuya et al., 1997; Guevara, 2004; Lara, 1993; Saxena & Chandra, 2011).

### 2.2.2 Bulkiness of ration

In formulating an animal diet, normally the ration weight or total ingredient weight is set in bulk such as 100 kilograms (kg) or 100 tonne because the ration is normally sold in weight (Waugh, 1951). By setting the ration in weight, farmers are able to estimate their budget to suit their farm size. They can also estimate the ration weight produced in order to avoid wastage or shortage. Therefore, almost all previous studies formulated animal diet in 100 kg for simplicity (Chakeredza, Akinnifesi, Ajayi, Sileshi, Mngomba, & Gondwe, 2008; Şahman et al., 2009; Thomson & Nolan, 2001). However, there are also studies that measured ration in percentages instead of in kilogram or tonne. These studies were conducted by Furuya et al. (1997) and Poojari and Varghese (2008). The bulk constraint was also considered in range by Mohr (1972) to increase the probability of achieving a feasible solution, which can be reduced up to 9.2% of the total cost.

### 2.2.3 Ration characteristic

Taste, smell and look are the ration characteristics that determine the final ration to feed the animal. These characteristics are closely related to the ingredient selection and ingredient restrictions, as observed by Swanson and Woodruff (1964) and Chappell (1974). Animal and plant based protein is one of the major factor that contributes to ration characteristic. Study in animal nutrition area is highly needed to identify the permitted range of specific plant-based protein to substitute animal protein sources. This is an important factor to attract animal attention to consume the ration produced (NRC, 2011).

### 2.2.4 Variability

Variability refers to the variation of nutrient composition and price of ingredients which fluctuates over time (Swanson & Woodruff, 1964). The price of the ingredients is based on the market price, while nutritive value depends on several factors such as weather conditions and environment. Guevara (2004), Roush, Cravener, and Zhang (1996), and Tozer (2000), are among the researchers that have taken variability into consideration when formulating animal diet.

### 2.2.5 User preference

User preference refers to a user judgment either to choose or not to choose the ingredients to be considered in the diet formulation. The preference allows the user to select available ingredients on farm to reduce feed storage. This feature can assist users who have the intention to test several combinations of specific ingredients. This factor was considered in a computerized diet system developed by Babić and Perić (2011), Şahman et al. (2009), and Thomson and Nolan (2001). Instead of preference on type of ingredient, user preference is also developed to cater nutrient relaxation problem (Lara & Romero, 1994) based on user perspective.

### 2.2.6 Environment

Environmental issue is related to nutrient excretion such as nitrogen and phosphorus from a digestible process of animal diet, which can pollute the environment. Several researchers (Moraes, Wilen, Robinson, & Fadel, 2012; Nguyen, Bouvarel, Ponchant, & van der Werf, 2012; Oishi, Kumagai, & Hirooka, 2011) who are concerned with this issue tried to minimize the nutrient excretion in order to sustain a green environment.

In formulating an animal diet that considers the above factors, literatures on previous techniques are presented in the next section to help identify the best possible technique to employ for this research.

## 2.3 Review of animal diet formulation techniques

This section highlights past studies that have been done on diet formulation problem, focusing on approaches to solving the problem in order to obtain the best results. They are algebraic (Afolayan & Afolayan, 2008; Gillespie & Flanders, 2009), optimization (Nguyen et al. 2012; Moraes et al. 2012), heuristics (Şahman et al., 2009; Pathumnakul, et al. 2011), and integrated approaches (Thammaniwit & Charnsethikul, 2013; Li & Jin, 2008). Algebraic approaches refer to the simple mathematics calculation on paper which is always used by students in an animal science nutrition study. Meanwhile, optimization technique consists of several mathematical programming techniques that aim to obtain a minimum cost diet.

Heuristics technique is a random technique applied to achieve the best-so-far solution. Finally, integrated approach is the hybrid between two or more methods. The whole range of techniques to solve diet formulation problem is depicted in Table 2.1 and also further presented graphically in Figure 2.1.

Table 2.1

*Classification of Diet Formulation Model by Solution Approaches*

| Approaches | Author |
|---|---|
| Algebraic | |
| • PS | Roush et al., 1996; Afolayan & Afolayan, 2008; Gillespie & Flanders, 2009 |
| • SAE | Afolayan & Afolayan, 2008; Gillespie & Flanders, 2009 |
| Optimization | |
| • LP | Waugh, 1951; Swanson & Woodruff, 1964; Nott & Combs, 1967; Rahman & Bender, 1971; Mohr, 1972; Chappell, 1974; Barbieri & Cuzon, 1980; Glen, 1980; Zioganas, 1981; Pierre & Harvey, 1986; Kock & Sinclair, 1987; Munford, 1989a; Munford, 1989b; O'Connor, Sniffen, Fox, & Miligan, 1989; Munford, 1996; Thomson & Nolan, 2001; Htun, Thein, & Tin, 2005; Olorunfemi, 2007; Chakeredza et al., 2008; Engelbrecht, 2008; Al-Deseit, 2009; Oishi et al., 2011; Udo, Ndome, Ekanem, & Asuquo, 2011b; Nguyen et al. 2012; Moraes et al. 2012; Piyaratne, Dias, & Attapattu, 2012 |
| • CCP | van de Panne & Popp, 1963; Pesti & Miller, 1993; Roush, Stock, Cravener & D'Alfonso, 1994; Roush et al., 1996; Pesti & Seila, 1999; Tozer, 2000; Roush & Cravener, 2002; Udo, Ndome, & Asuquo, 2011a |
| • QP | Chen, 1973 |
| • NLP | Guevara, 2004; Saxena & Chandra, 2011; Saxena, 2011 |
| • MCDM | Rehman & Romero, 1984; 1987; Lara & Romero, 1992; Lara, 1993; Lara & Romero, 1994; Mitani & Nakayama, 1997; Bailleul, Rivest, Dubeau, & Pomar, 2001; Tozer & Stokes, 2001; Zhang & Roush, 2002; Romero & Rehman, 2003; Castrodeza, Lara & Peña, 2005; Pomar, Dubeau, Létourneau-Montminy, Boucher & Julien, 2007; Peña, Lara, & Castrodeza, 2009; Babić & Perić, 2011 |
| Heuristics | |
| • TE | Forsyth, 1995; Afolayan & Afolayan, 2008 |
| • EA | Furuya et al., 1997; Şahman et al., 2009 |
| • Search-based heuristics | Pathumnakul, Ittiphalin, Piewthongngam, & Rujikietkumjorn, 2011 |
| Integrated | |
| • LP-based | Glen, 1986; Polimeno, Rehman, Neal, & Yates, 1999; Cadenas, Pelta, Pelta, & Verdegay, 2004; Alexander et al., 2006; Žgajnar, Juvančič, & Kavčič, 2008, 2009a; 2009b; Sirisatien, Wood, Dong. & Morel, 2009; Thammaniwit & Charnsethikul, 2013 |
| • Search-based | Li & Jin, 2008; Poojari & Varghese, 2008 |

## Abbreviation:

PS = Pearson's Square

SAE = Simultaneous Algebraic Equation

TE = Trial and Error

LP = Linear Programming

CCP = Chance constrained programming

QP= Quadratic Programming

NLP = Nonlinear Programming

MCDM = Multi Criteria Decision Making

EA = Evolutionary Algorithm

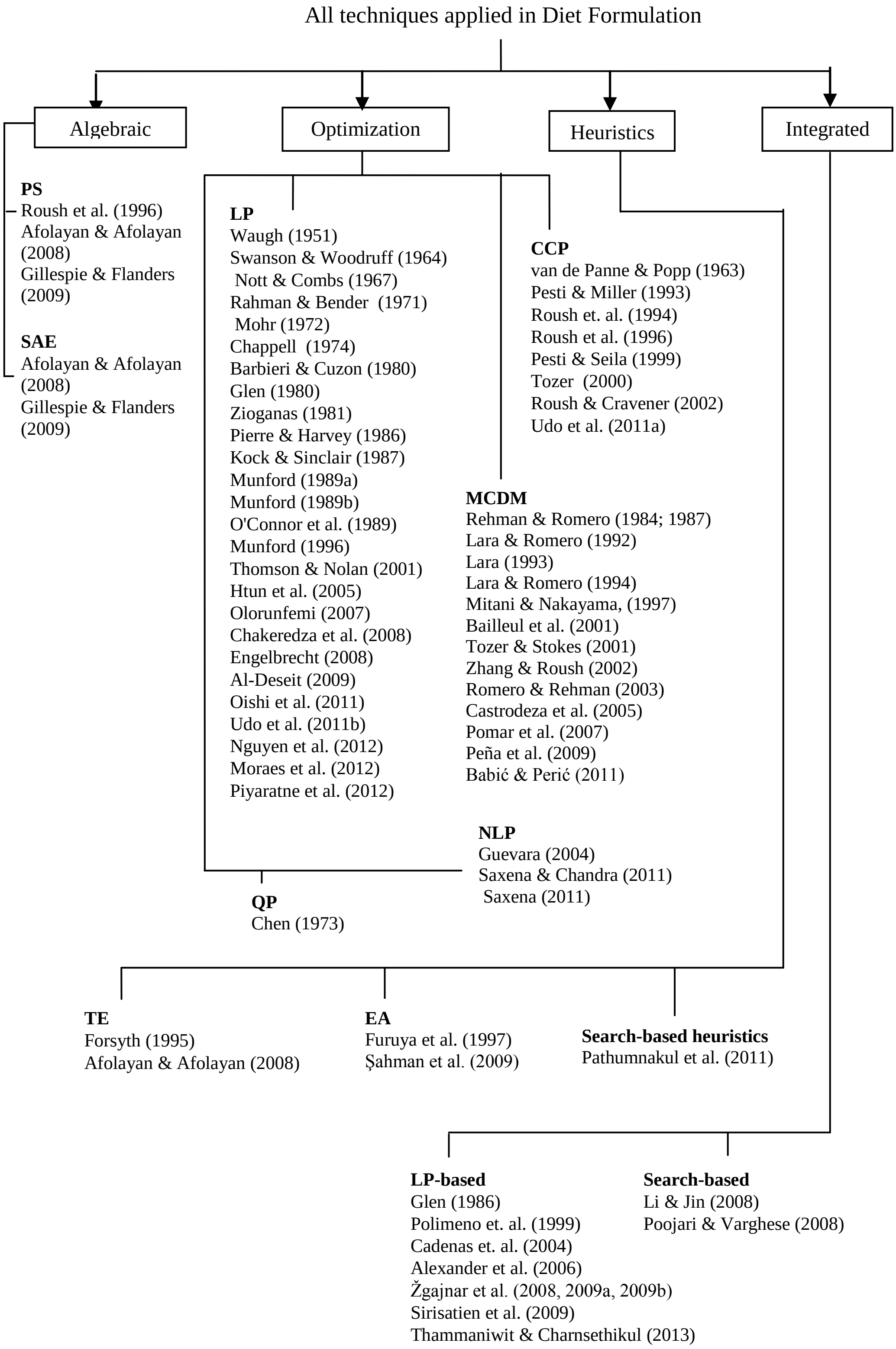



*Figure 2.1.* Graphical Presentation of the Classification by Solution Techniques

### 2.3.1 Algebraic approaches

In a diet formulation problem, there are two methods involving a simple calculation classified as algebra which is only appropriate for teaching purposes. These techniques are Pearson's Square and Simultaneous Algebraic Equations. Literatures on these techniques are discussed below.

#### 2.3.1.1 Pearson's Square

Pearson's Square (PS) is a simple technique involving several numbers in and around a square. The numbers represent the nutritional requirement for a specific nutrient (Wagner & Stanton, 2006). PS shows the proportions of two feed ingredients to be mixed together in order to obtain the percentage of the particular nutrients (Afolayan & Afolayan, 2008; Gillespie & Flanders, 2009). Due to this limitation, this technique is not possible to be used in a complex diet formulation problem, and is normally applied to balance protein requirements only. Roush et al. (1996) applied PS to balance corn and soybeans to meet 23% protein requirement. In their research, Roush et al. (1996) illustrated their work on CCP technique using PS to simplify their explanation.

#### 2.3.1.2 Simultaneous Algebraic Equations

An alternative method of PS is Simultaneous Algebraic Equations (SAE) method which is be able to balance two or more feed ingredients to achieve optimal nutrients value (Afolayan & Afolayan, 2008; Gillespie & Flanders, 2009). SAE is a condition where two or more unknown variables are related to each other through an equal number of equations (All about circuits, 2013). But this method can only balance for two nutrients at a time; hence it is not practical for solving a problem involving many nutrients.

### 2.3.2 Optimization approaches

Optimization refers to the selection of the best element from some set of available alternatives with regard to some criteria. Optimization problem involves maximizing or minimizing a function by systematically choosing the best value in a feasible region. Optimization is also defined as the process of attempting to find the best possible solution amongst all those available (Burke & Kendall, 2005). Optimization approach consists of various techniques including Linear Programming, Multi Objective Programming, Stochastic Programming, Quadratic Programming and Nonlinear Programming.

#### 2.3.2.1 Linear Programming

Linear Programming (LP) is a mathematical technique consisting of decision variables, objective function and model constraints (Taylor, 2007). Decision variables are mathematical symbols that represent the level of activity of a company. Objective function refers to the objective of the company either to minimize or maximize some value (i.e. minimize cost and maximize profit). Meanwhile, model constraints refer to the restriction faced by the company (Taylor, 2007).

In a farmed animal diet, Waugh (1951) was the first researcher who attempted to solve a diet formulation problem when he discovered that the LP method was best suited to solve an animal diet problem. The main purpose of the research was to investigate the capability of the LP technique in the area of animal diet. In the study, Waugh considered ten ingredients (corn, oats, milo maize, bran, flour middlings, linseed meal, cottonseed meal, soybean meal, gluten feed and hominy feed) and three nutrients, which were protein, calcium and phosphorus. The research objective was to minimize cost while satisfying animal nutritional needs. In general, the basic feed formulation model is as follows (Zhang, 1999):

$$f(x) = \min \sum_{i=1}^{n} (c_i \times x_i) \tag{2.1}$$

Subject to

$$\sum_{i=1}^{n} x_i = 1 \tag{2.2}$$

$$\sum_{i=1}^{n} a_{ij} x_i \geq b_j \tag{2.3}$$

$$u_i \geq x_i \geq l_i \tag{2.4}$$

Where $i$ the number of available ingredients ($i$=1, 2, ..., n)

$j$ the number of nutrient requirements ($j$ =1,2, ... , m)

$c_i$ the price of the $i^{th}$ ingredient

$a_{ij}$ the $j^{th}$ nutrient mean of the $i^{th}$ ingredient

$b_j$ the $j^{th}$ nutrient requirement

$x_i$ the percentage of the $i^{th}$ ingredient in the final ration

$l_i$ the minimum percentage of $i^{th}$ ingredient restricted in the ration

$u_i$ the maximum percentage of $i^{th}$ ingredient restricted in the ration

In his attempt, Waugh honestly admitted that he is not an animal nutritionist, but he was looking for a suitable method to facilitate an animal nutritionist job. This research has since led the way to many researchers to explore diet formulation problem. Waugh showed that the LP technique can be used to formulate diet for animal. However, in his study, Waugh (1951) did not consider the upper limit of nutrients which might be harmful to the animal's health when consumed over the recommended limit.

Even though LP as a tool to find an optimal solution in a diet formulation problem has received some reputation, it has several limitations. The first weakness is that the LP models assume that the nutrients levels are fixed, which means variability is not considered. Variability in ingredient's price causes nutritionists to formulate animal diet almost every day; this exercise wastes a lot of time and money. Furthermore, the computer performance in 1960's was not as efficient as today. Due to these reasons, Swanson and Woodruff (1964) introduced the dual statement in the LP model. This technique successfully reduced the computation time. However, the feed formulation still had to be done frequently. Chappell (1974) then overcame the variability problem and achieved the least cost diet through cooperation with several feed companies in the United Kingdom. Known as multi mix feedstock, price predictions in the coming few months were made based on the available ingredients at the companies. However, Kock and Sinclair (1987) criticized such method by arguing that the multi mix problem had to be run with a mainframe computer which is either expensive or time consuming. As a result, Kock and Sinclair divided the problem into a single mix model and solved it using LP.

The variability problem not only occurs in the price of the ingredients, but also in the nutrient composition of the ingredients. To cater this problem, Rahman and Bender (1971) inserted the element of probability into the LP model. The authors managed to modify the LP model and produced an encouraging solution with less time and complexity. However, the solution for iteration required a trial and error method (Roush et al., 1994). Pierre and Harvey (1986) then compared several existing rations with exact price based on the model developed by Rahman and Bender (1971). The exact price was very similar to the solution produced by the model. Another strategy to overcome the variability problem was proposed by Nott and Combs (1967, as cited in Roush et al., 1994) using LP with margin of safety. The technique was done by subtracting or adding 0.5 of the nutrient standard deviation from the nutrient mean. Nott and Combs claimed that this method increased at least 69 percent probability of meeting nutrient restriction (Roush et al., 1994).

Meanwhile, Glen (1980) and Munford (1989b) were concerned with overcoming LP limitation by tackling the nonlinear constraint that can include a ratio constraint in the LP model. They introduced a proportion of nutrients and then proposed a strategy to include the ratio constraint in LP using linearization procedure. This strategy was useful when taking into consideration the nutrient ratio using LP, but more calculation is required since differential calculus is essential in the procedure.

Oishi et al. (2011), Nguyen et al. (2012), and Moraes et al. (2012) included an environmental factor in the diet formulation model. LP was modified by Oishi et al. (2011) so that the objective of minimizing nitrogen and phosphorus excretions can be included in the model. The reason behind this strategy was because LP can only cater only one objective function. Oishi et al. claimed that their model was easier to be implemented compared with the multi objectives model designed by Tozer and Stokes (2001), and Castrodeza et al. (2005) (refer to Section 2.3.2.5). Afterwards, Nguyen et al. (2012) compared two models, with and without an environment factor, and then tried to find a compromising model between the economic factor and the environment factor. A study by Moraes et al. (2012) involved nitrogen and mineral excretion and also methane emissions. The result can guide feed formulators who want to include this factor since Moraes et al. considered environmental policies.

A comparison between techniques of Pearson’s Square (PS) and LP was made by Udo et al. (2011b) for an aquaculture diet. Udo et al. found that LP was more appropriate to use since the solution met all nine nutrient requirements. However, only five ingredients were taken into consideration and nutrient ratio was ignored. On the other hand, two studies conducted on shrimp by Barbieri and Cuzon (1980), and Htun et al. (2005) modelled several different formulated feeds to obtain the least cost feed. However, since there were many ingredient restrictions still under study, most of the ingredient restriction was considered as free value. Barbieri and Cuzon (1980) showed an encouraging solution when the cost reduction of almost 30% was obtained by LP.

Traditional LP models have been developed by several researchers to solve a case study (Zioganas, 1981; Olorunfemi, 2007; Engelbrecht, 2008; Al-Deseit, 2009; Piyaratne et al., 2012). Zioganas (1981),Olorunfemi (2007), and Engelbrecht (2008) suggested that most of the diet should be changed to get the least cost diets. Meanwhile, Al-Deseit (2009) and Piyaratne et al. (2012) put an effort to formulate a diet for broiler and poultry using local ingredients at their countries in order to minimize cost.

Instead of using LP, many feed formulation techniques are discovered by researchers with the aim to overcome LP limitations. In considering a variability factor, a number of researchers have applied stochastic programming technique which includes a probability element.

#### 2.3.2.2 Chance Constrained Programming

Chance Constrained Programming (CCP) is a stochastic optimization technique. CCP is formulated when a problem occurs in which some of the parameters are not constant (Prékopa, 2003). Assuming that the nutrients in the ingredients were not deterministic, van de Panne and Popp (1963) formulated diet for animal using this technique. The approach of this technique is based on probability concept. The probability value depends on the level of probability a researcher would like to achieve. van de Panne and Popp found that this technique was better than LP in terms of variability problem. However, the model required smart trial and error judgement to obtain a feasible solution or else no feasible solution might be met (Chen, 1973).

Meanwhile, in evaluating a CCP model in terms of variability, Roush et al. (1996) and Tozer (2000) compared CCP with LP with a margin of safety and proved that CCP provided a better solution almost for all cases. Meanwhile, Udo et al. (2011a) compared LP with CCP and found that CCP overcame the variability problem. Roush and Cravener (2002) conducted a research focusing specifically on variability among ten selected amino acids. They concluded that CCP was able to produce a better solution when variability was accounted in the model (Roush et al., 1996; Roush & Cravener, 2002; Tozer, 2000).

Roush et al. (1994) described the application of CCP to formulate an animal diet at Agway. The commercial feed formulation software known as Agway was delivered in 1991 and became the first computer program that uses CCP in the United States (Roush et al., 1994). The computer program was capable in producing faster solution as a trial and error technique was not necessary. Another computer program using CCP was developed by Pesti and Miller (1993) along with a reference book. The spreadsheet had the ability to edit constraints, but for new users the reference book was very useful to refer to the software precisely (Mathison, 1998). Soon in 1999, Pesti and Seila (1999) proposed a step on how to solve a linear and nonlinear diet formulation problem using a solver tool in an Excel spreadsheet.

In summary, CCP is a better approach than LP in terms of nutrients variability. The higher the probability value considered, the closer the optimal result will be, but the cost of the feed will increase. Hence, in formulation of an animal diet using CCP, a compromise solution between the probability value and the ration cost must be investigated since it could affect animals' health (Peña et al., 2009).

#### 2.3.2.3 Quadratic Programming

The characteristic of a Quadratic Programming (QP) is that it allows the objective function in the form of a quadratic subject to linear constraints. Chen (1973) proposed QP as an alternative to CCP with the aim to consider variability in a diet formulation modelling. In his research, Chen (1973) claimed that his model was suitable and easy to implement since no guessing was needed. However, he stated that the QP codes available at that time were not efficient for large problems.

#### 2.3.2.4 Nonlinear Programming

As the name suggests, Non Linear Programming (NLP) refers to the condition when problems fit the general LP format but do include a nonlinear function (Taylor III, 2007). The nonlinear function could be either in objective function or constraints. In order to tackle nonlinear constraints, Guevara (2004), Saxena (2011) and Saxena and Chandra (2011) carried out a study using NLP. Instead of considering nutrient ratio constraint, Guevara (2004) accounted for variability in price. NLP is a good technique as it considers a ratio constraint because in LP an approximation using linearization procedure is required (Saxena, 2011; Saxena & Chandra, 2011).

#### 2.3.2.5 Multi Criteria Decision Making

Multi Criteria Decision Making (MCDM) involves multiple conflicting criteria to be evaluated in making a decision. The strategy considers simultaneously optimizing two or more conflicting objectives subject to certain constraints. The solution is about satisfying all the objectives as prioritized. This characteristic becomes an advantage of this technique when a user (decision maker) has an option to choose the objective priorities based on his/her preference.

Goal Programming (GP) as part of the MCDM family was first applied by Anderson and Earle (1983) with the intention to solve nutrients imbalance problem for human diet. Then Rehman and Romero (1984) proposed a new framework in the diet formulation problem solving by considering many objectives and ranking the priorities of these goals. Other than minimizing cost, the study considered nutrient ratios and nutrient balance. Other research conducted by Rehman and Romero (1987) and Romero and Rehman (2003) stressed the use of penalty function into a diet formulation model as a way to overcome the strictness of the constraints in the LP formulation. This is another LP limitation, which means that no constraint violation is allowed. This weakness will normally lead to unfeasible solution in LP (Anderson & Earle, 1983; Munford, 1989a; Zhang & Roush, 2002).

Lara and Romero (1992) introduced the relaxation of nutrient requirements constraints in a diet formulation model. The relaxation allows the constraint to be violated since they believed that some violations did not harm an animal's health but resulted in a more economic diet. Subsequently, Lara and Romero (1994) extended their work by applying multiple interactive techniques known as step method (STEM). The interactive procedure refers to the user preferences regarding possible relaxations of the nutritional requirements obtained through a computerized dialogue. This technique successfully produced diet that satisfies the minimum nutritional imbalance based on the user opinion. However, the nutrient feed composition did not follow user preference and the user did not have the chance to choose the preferred ingredient.

Mitani and Nakayama (1997) agreed with Lara (1993) to consider the relaxation of nutrient constraint. Mitani and Nakayama (1997) then indicated the nutrient requirement as a soft constraint in the model. The main intention was to solve nutritional imbalance problem by improving the STEP method proposed by Lara and Romero (1994). Using satisfying trade-off method (STOM), they were able to show the ability of the method to adjust the upper value of nutrient requirements. However, the nutrient feed composition could not be edited and users still did not have the chance to choose the preferred ingredient. Meanwhile, Zhang and Roush

(2002) modelled on several objectives to compare the result based on objective priority. The findings showed that the priority of each goal had a great impact on the result obtained.

Tozer and Stokes (2001) and Bailleul et al. (2001) measured the environment factors by minimizing nitrogen and phosphorus excretion from dairy cows. Later, the same algorithm was altered by Pomar et al. (2007) who faced difficulty using Bailleul et al.'s (2001) model because small increase of feed cost would affect the nitrogen excretion seriously. Pomar et al. (2007) showed that the adequate level of phosphorus decreased when the feed cost increased.

Multi Objective Fractional Programming (MOFP) model was developed by Lara (1993) to include a ratio constraint in a diet formulation. By comparing MOP with MOFP, Lara concluded that the proposed model with a fractional objective could offer a better solution for a specific situation. Later, Castrodeza et al. (2005) included environmental factors in the MOFP model. The advantage of the MOFP approach is that it provides a wider variety of application (Kornbluth & Steuer, 1981). Afterwards, Peña, Lara, and Castrodeza (2009) continued Castrodeza et al.'s (2005) work by conducting a study using MOFP and considering variability issue. Peña et al. (2009) introduced an interactive process from user to make a decision on the permitted confront between the probability value of meeting the nutrients constraints and the ration cost as a way to overcome variability problem. The advantage of the interactive process was that it allows users to choose their preferred probability value based on user desires. This study however did not mention about user preference of ingredients as well as ration or total ingredient weight.

In 2011, a research on GP was conducted by Babić and Perić who gave an option to users to choose the preffered ingredients. In their study, Babić and Perić showed a solution when palatability factor was taken into account. The solution included two important ingredients, i.e., fish meal and soya hulls that gave a big impact on flavor and odor to attact animal into it.

Previous works show that the core advantage of MCDM is that it is able to handle multiple objectives simultaneously including nutrients variability and nutrients imbalance problems. MOFP offers more advantages as it could consider nutrient ratio in the model. The priority of its objectives selected depends on user preferences. However, reasonable goals and targets can only be obtained if the right target is chosen (Nijkamp & Spronk, 1978; Spronk, 1981), and choosing the right targets and weights is very difficult. This is because MCDM does not allow any trade-offs between goals. The first priority goal must be satisfied before the second one; not even one unit of the first priority goal can be sacrificed to obtain even a huge gain in the second priority goal (Zhang & Roush, 2002). Fortunately, many researchers have used MCDM in a diet formulation problem. So a researcher can refer to previous works to place the appropriate goal as a priority if he/she wishes to employ this technique. The introduction of an interactive procedure is more user friendly and it also serves as a learning tool for users as it enables them to experiment on different goals and priorities.

### 2.3.3 Heuristics approaches

Heuristics is a technique applied to find the best-so-far solution in a feasible region. Heuristics approaches can be divided into constructive heuristics that is Trial and

Error and Search technique applied in Artificial Intelligence. Two papers have been written on the Trial and Error technique, while the Search technique was initially proposed in 1997 to consider a nonlinear constraint which cannot be catered by LP technique (Furuya et al. 1997). To the best of our knowledge, only three papers have been written on search technique in a diet formulation problem that used EA and Search-based heuristics.

#### 2.3.3.1 Trial and Error

Trial and Error method (TE) is a technique of testing in a number of times to find the best solution (Longman, 2003). TE is one of the most popular methods for feed formulation for poultry (Afolayan & Afolayan, 2008). The formulation is done either manually on paper or by using computer spreadsheet such as Excel, Quattro Pro and Lotus123 (Forsyth, 1995). However, to gain the optimal result this method requires much time to spend especially when there are a lot of ingredients and nutrients that need to be considered. Currently, the Department of Fisheries (DOF) Malaysia is applying this method in formulating feed for aquaculture.

#### 2.3.3.2 Evolutionary Algorithm

As mention in Chapter One, EA is classified as a metaheuristics which is an upper level of heuristics (Talbi, 2009). Furuya et al. (1997) conducted a research using Evolutionary Algorithm (EA) with the aim to solve the nonlinear constraints which involved the ratio of ingredients. The study showed that EA is a good technique for diet formulation as a near optimal solution could be obtained even for a problem that has no apparent solution. In this research, Furuya et al. (1997) considered a minimum and maximum value of ingredient; however, almost all of the minimum values were considered as free value.

In a study conducted by Şahman et al. (2009), Genetic Algorithm (GA) was used to achieve least cost diet for livestock. Their GA experiments produced a good solution for a problem with a few constraints, which obtained zero penalty function value. The problems with many constraints resulted in some penalties value. However, the study by Şahman et al. (2009) did not consider a ratio constraint.

#### 2.3.3.3 Search-based heuristics

Search-based heuristics is an approach to find a near optimal solution for a complex problem in a reasonable time. However, the optimality of the solution cannot be guaranteed since the search is random without checking out all the possibilities (Burke & Kendall, 2005).

More recently, Pathumnakul et al. (2011) proposed a Search-based heuristics algorithm to solve several small-scale problems involving four to ten ingredients in each problem. They showed that the proposed heuristics technique is good at finding the best solution with a good computation time compared to a mathematical method.
There are several advantages of search technique especially GA-based in comparison to other techniques in diet formulation, as follows:

1. Diet formulation problem is a feed production planning (Dobos, Ashwood, Moore, &Youman, 2004) that can be classified as a NP-hard problem (Hung, Chen, Shih, & Hung, 2003; Pathumnakul et al., 2011). This problem aims at

the selection of a near-optimal mix of feed ingredients to satisfy some specific constraints. It constitutes a combinatorial optimization problem that is also deemed to be NP-hard in nature (Jiao, Zhang,& Wang, 2007). Hence, any search algorithm is appropriate to solve NP-hard as the processing time is faster than many other conventional technique (Pathumnakul et al., 2011).

2. The solution of EA might not be optimal since the concept of search is based on randomness, but it might satisfy all the constraints including for no solution problem (Furuya et al., 1997).
3. EA technique can easily incorporate a penalty function or can also be integrated with other technique (Fogel 1997; Sivanandam & Deepa, 2008). It is a suitable approach to solving rigidity in ingredients and nutrients constraints in a diet formulation problem.

### 2.3.4 Integrated approaches

Integrated approach is the combination of two or more techniques. In a diet formulation problem, most of the integrated techniques are basically based on LP and a few are search-based. Much research has been done in the diet formulation context which includes work related to the problem itself and also the solution techniques or methodologies being utilized.

#### 2.3.4.1 LP-based techniques

Glen (1986) and Polimeno et al. (1999) developed a model by integrating LP and Dynamic Programming (DP). LP model was applied to minimize total cost of feed ingredients while DP was applied to determine the optimal series of animal weight changes for a specific period. This approach is appropriate to determine the optimal ration in the condition of involving phase.

Arguing that it is impossible to have a perfect knowledge of all or some of the data, Cadenas et al. (2004) discovered the application of fuzzy concept in a cattle farm. The authors intended to minimize cost by giving the appropriate quantity of ration to be eaten without being wasted. They considered nutrient and ingredient restriction based on animal weight but lacking in nutrients ratio. This approach is one of the superior contributions towards diet formulation area that has a great potential for further investigation.

Žgajnar et al. (2008, 2009a, 2009b) combined LP and weighted GP (WGP) in order to produce an optimal ration cost with balanced nutritional requirements. They were primarily concerned with overcoming the LP drawback that is LP rigidity. At first, LP was applied with the aim to estimate the final cost in the ration. Then, the cost was placed into the second model which was WGP with penalty function as the desired value to achieve. These studies considered nutrient and ingredient restriction but ratio of nutrient was not taken into account.

Minimizing cost and maximizing profit is the main objective of every business. Feeding schedule was discovered by Alexander et al. (2006) along with minimizing feed cost in order to maximize profitability. The optimal feeding schedule was obtained by Nonlinear Optimization (NO) technique, while LP was applied to find the minimum feed cost. Nutrient and ingredient restriction as well as ratio of nutrients were also considered. Alexander et al.'s (2006) study was extended by Sirisatien et al. (2009), who took into account variations in cost and price.

Recently, Thammaniwit and Charnsethikul (2013) combined LP with Bender's Decomposition method. They aimed to tackle the variability problem that often limits the LP technique. The researchers showed that this technique can produce an optimal solution. However, instead of variability, other factors that influence diet formulation were ignored.

#### 2.3.4.2 Search-based technique

Li and Jin (2008) recommended combining GA and fuzzy optimization method to overcome fuzzy problem in Cadenas et al.'s (2004) study. They included nutrient ratio as one of the constraints to be achieved. The advantage of this algorithm as compared to Cadenas et al.'s (2004) was that it is able to tackle both linear and nonlinear constraints.

Poojari and Varghese (2008) integrated GA and MCS to solve uncertain problems in many fields. GA was applied to cater a nonlinear nature while MCS to cater a stochastic nature. They showed that the hybrid method was comparable with CCP. However, user preference was not considered in their study.

## 2.4 Computer developments

The advancement of computer development simplifies the process of decision making with flexibility of editing the ingredients and nutrient constraints. Since 1980's many decision support system (DSS) have been developed that offers different quality features especially user friendly interface and flexibility. User friendly is an important element to connect the decision maker with little or no knowledge in operational research technique with the generated mechanism. On the other hand, flexibility allows users to edit, add or delete constraints that consist of ingredients and nutrient restrictions. Chakeredza et al. (2008), Guevara (2004), Munford (1989a), Munford (1996), O'Connor et al. (1989), Pesti and Miller (1993), Pesti and Seila (1999), Şahman et al. (2009), and, Thomson and Nolan (2001), were among those who have developed the DSS for user benefit.

DSS is developed for different purposes. For instance, Thomson and Nolan (2001) designed a spreadsheet known as UNEform for teaching purposes. Hence, UNEform was created to be transparent in calculation because Thomson and Nolan criticized that many commercial feed formulation softwares available in the market hide it from users. Another system was developed by Chakeredza et al. (2008) with the intention to help small scale farmers who utilize available ingredients on the farm. They demonstrated how to develop the spreadsheet step by step. The spreadsheet also has the ability to select feed ingredient and edit nutrient and ingredient restriction.

Another advantage of DSS or computerized system is one can put user preference factor under consideration. This factor is taken into account by researchers such as Thomson and Nolan (2001), and Şahman et al. (2009). User preference factor gives a great impact to users as they are able to choose any ingredients to their preferences. The available ingredients in the farm become the priority to consider in the diet formulation since it can help farmers reduce their cost and increase profit.

## 2.5 Differences among Evolutionary Algorithms models of diet formulation

Since a few EA models are available, this section lists all previous EA models that have been developed for diet formulation. The similarities and differences of each model are highlighted so that the contribution of our model can be seen clearly. Table 2.2 summarizes the differences.

Based on Table 2.2, our objective is a combination of Furuya et al.'s (1997) and Şahman et al.'s (2009) that is to obtain minimum cost diet for shrimp that consists of nutrient ratio, ingredient and nutrient restriction and also user preference of ingredients. In addition, our research includes user preference of total ingredient weight with the aim to avoid wastage or shortage of ration.

In terms of research complexity, we consider more constraints compared to other four researches. It consists of sixteen single nutrients, two combinations of nutrients and one ratio of nutrients. Ninety-one types of ingredients are taken into consideration. In this research, EA with enhanced search operators is proposed as a methodology to achieve the objective. The methodology will be discussed in the next chapter.

Table 2.2

*Summaries and Scope of Four Main Studies Using EA in Diet Formulation Area*

| **Description** | **Furuya et al. (1997)** | **Şahman et al. (2009)** | **Li and Jin (2008)** | **Poojari and Varghese (2008)** |
|---|---|---|---|---|
| Objective | To solve the nonlinear constraints which involved the ratio of nutrients and obtain minimum cost ration for livestock | To obtain minimize cost diet for poultry and cattle | To model the impact of fuzzy factor in diet formulation and obtain minimum cost ration for pig | To model the impact of variability factor in diet formulation and obtain minimum cost ration for cattle |
| Other issue taken into account | Balanced in nutrient involving nutrient ratio, ingredient and nutrient restriction. However, many ingredient restrictions are put as free value | Balanced in nutrient involving nutrient and ingredient restriction. Consider on user preference factor. | Balanced in nutrient involving ingredient and nutrient restriction | Balanced in nutrient involving nutrient restriction |
| Number of ingredient and nutrient | 20 ingredients, 12 single nutrients and 1 ratio nutrients | *Data 1:*<br>11 ingredients and 10 single nutrients<br>*Data 2:*<br>7 ingredients and 8 single nutrients | 3 ingredients and 2 nutrients | 4 ingredients and 2 nutrients |
| EA Operators | New crossover and mutation is introduced | Basic GA with elitism procedure and penalty function. | Integrate GA with fuzzy | Integrate GA with MCS |

## 2.6 Summary

Algebraic, optimization, heuristics and integrated approaches are the existing approaches used for diet formulation. An algebraic approach is not applicable since it is not capable to cater complex problem. Optimization approaches are the most favourable technique in diet formulation field. However for NP-hard problem, long time duration is required. In contrast, heuristics approach can be considered as a new branch of research method used by previous research in this field. There are many existing search methods such as Tabu Search, Simulated Annealing, and Ant Colony Optimization, which may be employed to solve diet formulation problem. In fact, EA with enhanced search operators such as initialization and crossover may also be considered since not so many GA-based techniques have been used.

Rigorous studies in diet formulation show that there are many elements that have been investigated using various techniques to obtain a useful result. Each research contributes new information in diet formulation. These elements are balanced in nutrients, bulkiness of ration, ration characteristic, variability, user preference and environment. The difference of this study with others is we considered on four elements. They are balanced in nutrients, bulkiness of ration, ration characteristic and user preferences. Variability element is not considered, however, users may edit nutrient and price value in interface to meet the actual value if there are changes in real situations. Environment factor is also not considered directly, despite being considered by previous research. However, one of the purposes on this research is to reduce ration waste by investigating specific target weight. Therefore, this study introduces preferred bulky element into the diet formulation model. Farmers or nutritionists do not have the freedom to fix the weight they require if the weight is set at 100 pounds or 100 kg. They may instead need 120 kg for their farm, or they may also require a two tonne ration for a large scale farm.

A DSS or computer generated system is a must have because it facilitates the process of animal diet formulation. It is preferred that the computer system is flexible, user friendly and has user preferences. Therefore, this research addresses these requirements with the approach of Evolutionary Algorithm as highlighted in the next chapter.

# CHAPTER 3

# THE CONCEPT OF EVOLUTIONARY ALGORITHM

As discussed in the previous chapter, heuristics is a method that still can be explored and developed. Therefore, this chapter mainly discuss the concept of EA to understand the approaches associated with it. Firstly, this chapter discusses the taxonomy of heuristics methods to capture the broad picture of different techniques in heuristics family and the position of EA in the family. Then, Section 3.2 until 3.5 reviews the fundamentals of evolutionary technique especially the GA-based. Section 3.6 discusses the constraint handling technique to solve various complex problems whereas Section 3.7 concludes the chapter.

## 3.1 Taxonomy of heuristics approaches

A heuristic search method is also known as an approximation method due to the concept of randomness in order to search for the optimal solution (Das, Abraham, & Konar, 2009). One can say that the solution from this method is a *near optimal* solution (Babu & Murty, 1993; Bauer, 1994). Heuristics approaches are divided into two: deterministic and probabilistic. Deterministic is also divided by two: classical and modern. Classical approaches consist of Greedy Search, Hill-Climbing, and Variable Neighbourhood Search while modern approaches consist of Tabu Search. Meanwhile, probabilistic approaches are categorized by a number of solutions either single-based or population-based. Single-based solution consists of Simulated Annealing and Stochastic Hill-Climbing while population-based solution contains Evolutionary Computing techniques. Two categories in Evolutionary Computing techniques are Evolutionary Algorithms and Swarm Intelligence Algorithms.

Evolutionary Algorithms (EA) consist of Genetic Algorithm, Evolutionary Programming, Evolution Strategy, Genetic Programming and hybrids of any EAs technique. Meanwhile, Ant Colony Optimization, Cuckoo Search, and Artificial Bee Colony Algorithms are among the techniques in Swarm Intelligence Algorithms. Figure 3.1 shows the taxonomy of heuristics graphically.

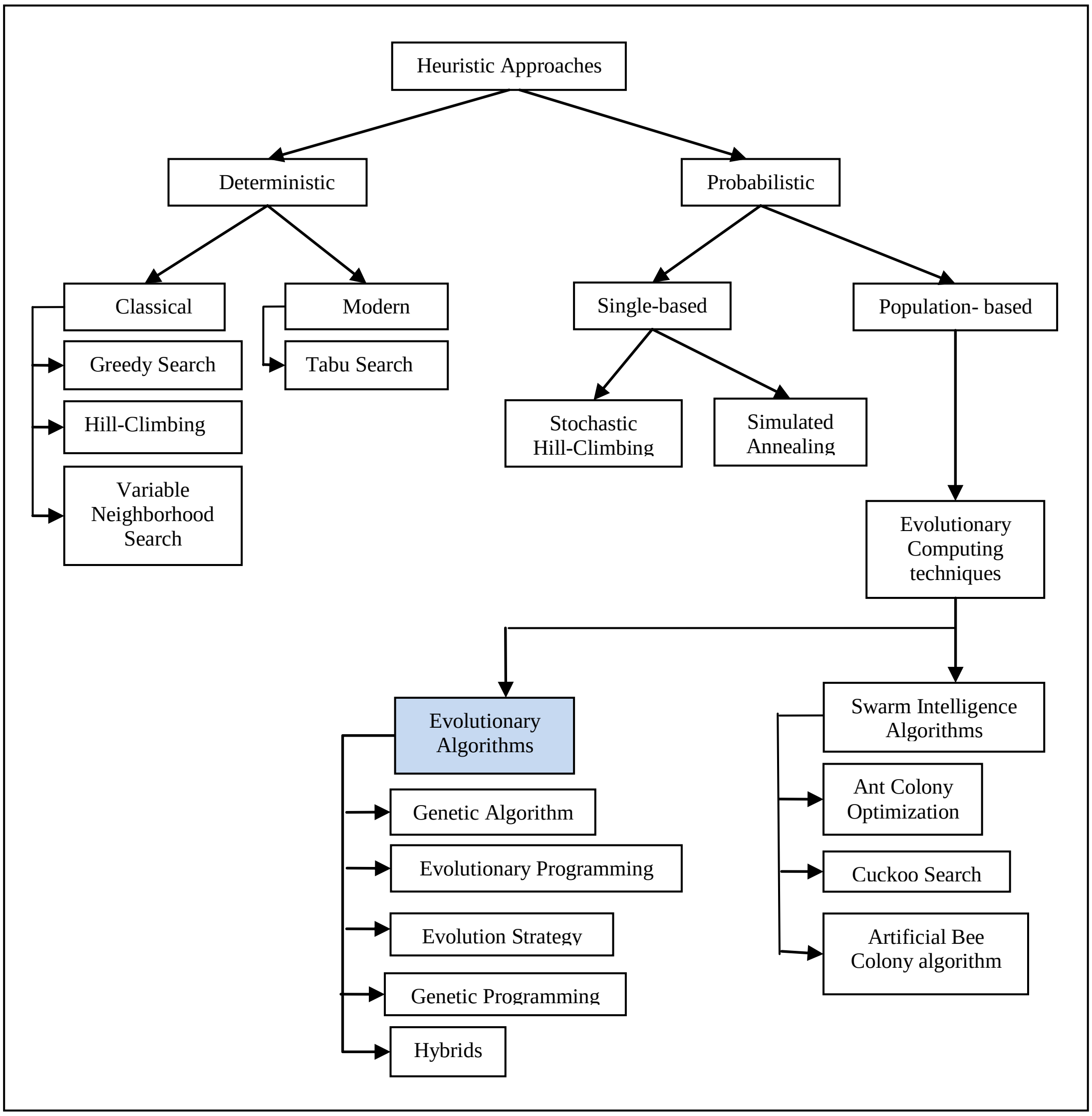



*Figure 3.1*. Evolutionary Algorithm in the Taxonomy of Heuristics Methods as Adapted and Enhanced From Das, Abraham & Konar (2009)

Among all methods shown in Figure 3.1, the hybrid technique and Swarm Intelligence are developed rapidly in present time. For instance, the combination of Genetic Algorithm with Simulated Annealing (Yu, Fang, Yao, & Yuan, 2000; Guoshuai, Kan, Ran, & Jiao, 2012), the combination of Swarm Intelligence such as Ant Colony Optimization with Genetic Algorithm (Rossi & Boschi, 2009; Zhao, Wu, Zhao, & Quan, 2010) and the modification on the technique itself such as enhancement of the search operator (Deep & Mebrahtu, 2011). Meanwhile, Swarm Intelligence is developed with many new proposed algorithms such as Cuckoo Search (Yang & Deb, 2009), and Artificial Bee Colony Algorithm (Karaboga & Basturk, 2007, 2008).

As is known EA is a population-based method which is classified under an Evolutionary Computing technique. EA is widely used in many fields due to the robust adaptation to the environment (Fogel, 2000a). Researchers who have employed EA in their studies include Bhanja, Mahapatra, and Roy (2013), Chang, Chen, Tiwari, and Iquebal (2013), Hecker, Hussein, Paquet-Durand, Hussein, and Becker (2013), and Zhang, Xu, and Gen (2013), to name a few.

Among other evolutionary methods, Genetic Algorithm (GA) is the most commonly used due to its successful achievement when applied to many real world problems (McCall, 2005). GA was invented by Prof. John Holland in 1975 and continued to be investigated by Prof. David Goldberg (Goldberg, 1989; Mitchell, 1996). For this reason, the technique employed in this research is EA which is based on GA. The following section will discuss the fundamentals of EA.

## 3.2 Fundamentals of Evolutionary Algorithm

Evolutionary Algorithms consists of several techniques: Genetic Algorithm, Evolutionary Programming, Evolution Strategy, Genetic Programming, and the hybrids technique. These techniques have one fundamental commonality that is they involve the reproduction, random variation, competition and selection of competing individuals in a population (Fogel, 2000a). These processes form the evolution which is known as the neo-Darwinian paradigm that is also the most widely accepted collection of evolutionary theories (Fogel, 2000b). Some of the prominent characteristics of the neo-Darwinian paradigm have been summarized by Mayr (1988, as cited in Fogel, 2000b) as follows:

1. The individual is the primary target of selection.
2. Genetic variation is largely a chance phenomenon. Stochastic processes play a significant role in evolution.
3. Genotypic variation is largely a product of recombination and 'only ultimately of mutation'.
4. 'Gradual' evolution may incorporate phenotypic discontinuities.
5. Not all phenotypic changes are necessarily consequences of ad hoc natural selection.
6. Evolution is a change in adaptation and diversity, not just a change in gene frequencies.
7. Selection is probabilistic, not deterministic.

In addition, as mentioned by Goldberg (1989), EA is different from optimization and other search procedures in three ways:

1. EA works with a coding of parameter set, but not the parameter themselves.

2. EA searches from a population of possible solution, not a single point of solution.
3. EA works based on objective function value, not the derivatives or other knowledge.

### 3.3 Advantages of Evolutionary Algorithms

Fogel (1997, 2000) lists several advantages of EA over many search or optimization algorithms. These include the following:

1. EA can be applied to any optimization problem. The flexibility of EAs also allows this method to be applied to discrete, continuous and mixed integer problems. EAs are being applied to a certain number of areas in which computers have not been used before (Beasley, 2000).
2. EAs have the ability to solve problems by automated routines when there are no human experts to learn from experience. After running EA for several times with little information about the problem domain, a near optimal solution can be obtained.
3. The robustness of EA extends to a broader field of applications (Schwefel, 2000) that includes routing, scheduling, packing, and classification (Beasley, 2000).
4. EAs provide a framework that is easy to understand and handle (Schwefel, 2000). Therofore, EAs can be used for further specialization and hybridization with other methods.

### 3.4 Structure of Evolutionary Algorithm

EA is a search method based on natural evolution that has been introduced by Darwin. ‘Survival of the fittest’ in Darwin’s theory refers to the competition in getting food among individuals (Goldberg, 1989). High fit individuals will survive and evolve to the next generation, while less fit individuals tend to die. The combination of superior individuals might produce better children than the parents themselves.

In EA theory, the characteristics of each individual are evaluated by fitness value (Bäck, 2000) which represents whether the individual is good or bad. The process of EA starts by selecting two individuals as parents from a set of possible solutions based on their fitness value. These parents then will go through several operators such as crossover and mutation to produce a child or offspring. This process is repeated over a few generations while the high fit individuals with good characteristics will survive. At last, a near optimal solution is discovered after going through all the processes as shown in Figure 3.2.

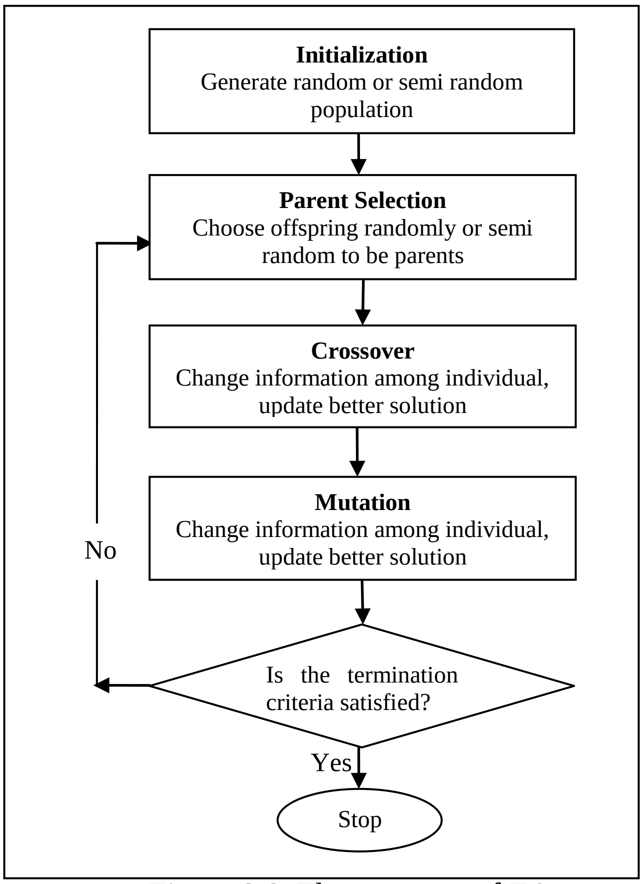



*Figure 3.2.* The structure of EA

The basic steps involved in EA include initialization, selection, crossover and mutation as shown in the Figure 3.2. However, unlike GA, the search operator may not be included in the whole EA stages. But the operators (i.e. mutation) may also be included more than once. The step-by-step procedure as shown in Figure 3.2 will be discussed in the following section. However, the first step in implementing EA is to define the solution representation carefully in order to understand how the end result will appear.

### 3.4.1 Representation of solution

Most of the processing time in EA is spent on evaluating population. This is the reason why the representation structure is very important (Golub, 1996). The representation could be in many forms such as an array of bits, a number, an array of numbers, a matrix or a string of characters (Golub, 1996), known as chromosomes. It brings information or characteristics about the solutions.

Several encoding representation types are binary, real value, and permutation. The way to represent the solution depends on the specific problem to solve (Deb, 2000a). For instance, an order-based problem such as a travelling salesman problem is more appropriate with permutation encoding as the objective is to search for a near optimal ordering of elements.

Of all encoding types, the most frequently used encoding technique is binary encoding (Deb, 2000a) due to its classical and simple approaches (Bäck, 2000). Binary encoding is a simple GA approach proposed by Holland and his students using zero and one. One example of a problem that is suitable to solve using binary encoding is knapsack problem which requires only two decision variables; bring along the stuff or not to bring.

Real number encoding using real value is introduced with the objective to bring EA closer to the problem space. Michalewicz (1994) experimented that, in developing optimization problems with continuous variable, real value is more appropriate to use since real number encoding provides more precision, faster and more consistent in each run than binary encoding. He also pointed out that using binary coding requires much computing time especially in large domains. Real value encoding is more appropriate to use in a continuous domain such as problems in medical and product mix (Deb, 2000a; Deep, Singh, Kansal, & Mohan, 2009; Sung, II-Hwan, V. Mani, & Hyung, 2008).

### 3.4.2 Solution initialization

The initial population of solution can be generated in random or semi random (Ramli, 2004). Random procedure applies when the solution is generated randomly without any requisite. Semi-random initialization involves some condition or any heuristics technique to ensure good individuals are selected in an early stage so that a feasible solution can be obtained faster.

### 3.4.3 Selection

Selection or reproduction is one of the important operators used in EAs. Basically, the purpose of a selection process is to choose better solutions to be parents for the next step and delete the remaining worse solution (Deb, 2000b). Sivaraj and Ravichandran (2011) reviewed several selection operators in EA. They include Roulette Wheel, Deterministic Sampling, Linear Ranking Selection, Binary Tournament Selection, and Range Selection. Different selection mechanisms work well for different problems (Sivaraj & Ravichandran, 2011). Thus, the most suitable selection mechanism has to be chosen for a specific problem to increase the optimality of the solution. Even though there are many selection mechanism that have been proposed, this section highlights the three most widely used selection mechanism. They are Roulette Wheel Selection (RWS), Tournament Selection, and Ranking Selection. These selection mechanisms are favourable to many researchers because they are simple and have been proven to produce an acceptable solution.

#### 3.4.3.1 Roulette Wheel Selection (RWS)

RWS was proposed by Holland in 1975 and has been used widely in the application of EA. RWS is quite an established technique and it becomes the simplest selection operators that are based on the concepts of proportionate (Deb, 2000b).

Conceptually, the fitness value of each individual in the population corresponds to the area on the Roulette Wheel proportion. Then, the Roulette Wheel is spun; a solution marked by Roulette Wheel pointer is selected. Higher fitness with a bigger area is likely to have more chances to be chosen. The segment size and selection probability remain the same throughout the selection stage (Chipperfield, 1997).

The advantage of the RWS technique is that it gives no bias with unlimited spread (Chipperfield, 1997). However, one of the disadvantages of RWS is that it cannot handle negative fitness values due to the proportionate concept. Also, RWS cannot handle minimization problem directly. However, this limitation can be overcome by making some transformations into equivalent maximization problem (Deb, 2000b).

#### 3.4.3.2 Tournament Selection

Competition among a group of parents is the basis of Tournament Selection operator. Measurement of fitness of solution is made among all parents and the parent having best fitness is selected. The term ‘Binary Tournament’ refers to two tournament sizes which is the simplest form of Tournament Selection (Blickle, 2000; Deb, 2000b). Binary Tournament Selection starts by selecting two individuals randomly. Then, a fitness value between these individuals is evaluated. The one that has better fitness is chosen.

One advantage of Tournament Selection is that it can handle either minimization or maximization problems without any structural changes. In addition, negative value is allowed without any restriction. Due to the strength of Tournament Selection, it is possible to further explore this technique.

#### 3.4.3.3 Ranking Selection

Ranking Selection is similar to Proportionate Selection. The individual is arranged either in an ascending or descending manner depending on the type of problem either minimization or maximization. The ranking of the individuals starts from 1 to $N$, according to their fitness value.

Similar to Tournament Selection, Ranking Selection can handle minimization or maximization problem as well as negative values. Another advantage of this selection operator is that it can avoid fast convergence (Grefenstette, 2000). However, the disadvantage of this technique is that it might slow down the selection pressure since the difference between the best individual with others is small (Mitchell, 1996).

### 3.4.4 Crossover

The main distinguished feature of a EA is the use of crossover (Mitchell, 1996). Recombination or crossover combines part of two or more chosen parents to create a new and better solution. However, the routine crossover takes two parent strings called parent 1 and parent 2 and generates two offspring strings called child 1 and child 2 (Herrera, Lozano, & Sánchez, 2005).

As the most important search operator, many techniques have been proposed to accomplish the idea of crossover. As a result, crossover has incorporated a special feature such as statistical element (Arithmetic Crossover) (Michalewicz, 1994), fuzzy element (Extended Fuzzy Crossover) (Herrera & Lozano, 2000), and natural observation (Queen-Bee Crossover) (Azeem & Saad, 2004). The special feature is inserted into the crossover and the name of the crossover is given based on the inserted feature. For instance, in Queen-Bee Crossover the behaviour of a queen bee as a dominant figure is adopted as parent 1. Parent 2 is then chosen randomly from other individuals.

There is no single crossover type that will go well with every problem (Herrera et al., 2005). For instance, Multi-Point Crossover may produce an acceptable solution for menu planning problem but produce a bad solution for timetabling problem in certain condition. However, the most common crossover operator used in the application of EA is One-Point Crossover, Multi-Point Crossover, and Uniform Crossover.

#### 3.4.4.1 One-Point Crossover

In One-Point Crossover, a single crossover position within the chromosome length is selected either randomly or is fixed at a certain point. Then, two new individuals are created by swapping all bits after that point (Booker, Fogel, Whitley, Angeline, & Eiben, 2000). One-Point Crossover is an established type of crossover; however, it is possible to further explore on this mechanism especially for real number representation problem.

#### 3.4.4.2 Multi-Point Crossover

Multi-Point Crossover is an extension of One-Point Crossover where two or more points are chosen at random as crossover position. In Multi-Point Crossover, Two-Point is the most frequently used in the application of EA (Booker et al., 2000) by swapping the corresponding segments from the two parents defined by the two chosen points. However, a population may become homogenous after many generations (Spears & De Jong, 1990) especially for short structure.

#### 3.4.4.3 Uniform Crossover

Instead of One-Point and Multi-Point Crossover, Uniform Crossover is also a favourable type of crossover to choose due to its simplicity. Uniform Crossover mechanism randomly swaps individuals between parents one and two without using any cutting point (Spears, 2000). The number of crossover point is not fixed in advance but all genes have the same chance of being disrupted (Ramli, 2004).

### 3.4.5 Mutation

Mutation is another important search operator in EA. Crossover operates on two parental chromosomes, while mutation locally but randomly modifies a solution. Mutation works on only one chosen chromosome as the main purpose of mutation is to avoid local minima by preventing the population of chromosomes from becoming too similar to each other that might affect the evolution process (Eshelman, 2000). Mutation task is of fine-tuning the solution as normally necessary in a constrained optimization problem. Again, there are many variations of mutation; however the implementation of mutation is closely related to the type of encoding. Instead of encoding type, some literature shows that mutation is designed for constrained or unconstrained problem such as Uniform Mutation (Michalewicz, 1992), and Power Mutation (Deep & Thakur, 2007).

#### 3.4.5.1 Boundary Mutation

Michalewicz (1992) proposed Uniform Mutation where a gene is replaced with a random value between the constraint range (lower and upper bounds). A special case of Uniform Mutation is Boundary Mutation (Michalewicz & Schoenauer, 1996; Hasan & Saleh, 2011) in which a gene is replaced by either its lower bound or upper bound value.

#### 3.4.5.2 Power Mutation

Power Mutation is introduced by Deep and Thakur (2007) for constrained problem with real number representation structure. Power Mutation is the enhancement of Uniform Mutation designed based on a random concept and makes use of the both lower bound and upper bound of constraint. The advantage of using constraint boundaries is that we might get a number of possible solutions within the constraints range. Therefore, Power Mutation can be used in EA without incorporating other constraint handling technique. The strength of Power Mutation is that it is controlled by its index; a small index value is expected to produce less perturbance in the solution and large index values are expected to achieve large diversity.

### 3.4.6 Reproduction procedures

Once all steps in EA, i.e., initialization, selection, crossover and mutation, are done, it means that one generation is completed. This process continues to the next generation by either replacing the whole population with a new one or only a few less fit individuals are replaced. Basically, the reproduction operators control how EA creates the next generation. Reproduction is also a selection operator in which a number of good solution is chosen for the next generation.

Generational Reproduction concepts apply when the whole population is replaced by the new individual in every generation. Generational Reproduction concept is known as non-overlapping system (De Jong &Sarma, 1993). On the other hand, Steady-State Reproduction means only a few individuals are replaced by offspring in each generation. Steady-State is an overlapping system as parents and offspring are competing for survival (Lozano, Herrera & Cano, 2008).

### 3.4.7 Elitism

Elitism is another operator in EA that copies the best chromosome or a few best chromosomes to the next population (Lozano et al., 2008). This operator is introduced by De Jong in 1975 in his PhD thesis (Golberg, 1989). The advantage of elitism is that it can increase EA performance as it prevents losing the best found solution (López-Pujalte, Bote & Anegón, 2002; Sharief, Eldho & Rastogi, 2008).

### 3.4.8 Termination criteria

Termination is the criterion by which EA decides whether or not to continue searching or stopping the search process. In EA, several criteria are suggested as a stopping criterion. Some common criteria are generation numbers, computation time, or when a minimum criterion is reach.

Generation number is a termination method that stops the evolution when the predetermined number of generation is specified by a user. In diet formulation, Şahman et al. (2009) applied this termination method as well. Instead of generation number, a user is able to set the maximum computation time as termination criteria. Besides, sometimes a condition such as “stop when there is no improvement during last 10 generations” can be used to stop the evolution (Aytug & Koehler, 2000).

Another method to use is by fixing specific fitness value (Aytug & Koehler, 2000). For minimization problem, the search process is stopped when a fitness value is less than the user’s specific fitness threshold. On the other hand, the search process is stopped when the fitness threshold is greater than the user specific fitness value for maximization problem.

### 3.5 Fitness evaluation

Objective function reflects how fitness value is evaluated. Maximization problem requires that bigger fitness value is better, whereas minimization problem prefers less fitness value (Kinnear, 2000). Objective function can also be evaluated based on penalty values, where in minimization problem, a less penalty value is better and vice versa. For further explanation and mathematical modelling of the objective function and constraints, one may refer to Section 4.3.5 and 4.3.6.

### 3.6 Constraint handling technique

Most of the problems exist in this world are constrained problem such as the knapsack problem, vehicle routing problem, scheduling problem, timetabling problem, and diet formulation problem. A search space consists of a feasible and infeasible region that we have to deal with (Michalewicz, 2000b). EA is capable of handling constraints as solutions are arrived at each point in the search. However, a constraint violation is very difficult to resolve as the chances to obtain a feasible solution are very slim (Michalewicz, 1994). For instance, in the area of timetabling where a lot of research has been conducted using GA-based, the use of standard GA does not yield practical timetables because too many constraints were violated (Wilke, Gröbner, & Oster, 2002; Yigit, 2007). Therefore, several constraint handling techniques are proposed with the most favourable technique is discussed below.

### 3.6.1 Penalty functions

Penalty functions have been discovered for a long time for constrained optimization problem (Smith & Coit, 2000). The implementation of penalty function in EA is efficient. For instance, Şahman et al. (2009) adopted penalty function in diet formulation problem and produced valuable result. Other researchers (Lara & Romero, 1994; Rehman & Romero, 1987; Žgajnar et al., 2009a) incorporated penalty function with the same objective i.e. to obtain a feasible solution. The implementation of penalty function was found to successfully increase the chances of acquiring a feasible region.

### 3.6.2 Repair algorithm

Basically, there is no standard way to use repair algorithms. Normally, one can use special heuristics design for a specific problem since repair algorithms is problem dependence (Michalewicz, 2000c). Salcedo-Sanz (2009) reviewed 137 papers and concluded that hybrid approaches with repair heuristics obtained better results than other constraint handling techniques. This finding is supported by Coello (2002), who claimed that in a combinatorial optimization problem, repair algorithms is the most appropriate to use. Yigit (2007) demonstrated that the incorporation of repair operator can successfully reduce the penalty function value in timetabling problem and decrease the computing time as well.

Many researchers have experimented on hybrid evolutionary systems as well as repair algorithm based on three principles identified by Davis (1991). The principles are as follows:

1. Use the current encoding: use the current algorithm's encoding technique in the hybrid algorithm
2. Hybridize where possible: incorporate the positive features of the current algorithm in the hybrid algorithm
3. Adapt the genetic operators: create crossover and mutation operators for the new type of encoding by analogy with bit string crossover and mutation operators. Domain based-heuristics may incorporate as operators as well.

There exist several heuristic algorithms either better or worse for a specific problem. The hybrid EA consists of a system which extends the paradigm of EA by incorporating additional features (local search, problem-specific representation and operators, special heuristics). Some of the features are incorporated for initialization purposes while some of these algorithms involve changes in solution encoding (Michalewicz, 2000a). Otherwise, many algorithms exist by combining existing knowledge in search operators with the problem domain (Davis, 1991).

Based on the principles in designing hybrid Evolutionary Algorithm recognized by Davis (1991), several existing methodologies has been hybridized with EA. One of the most frequently methods hybridized with GA as a repair method is Local Search. Hybridization of Local Search (LS) and GA is very common in various domains which are considered as ‘intelligent mutations’. This concept is known as Memetic Algorithm. On research conducted by Ramli (2004) using Memetic Algorithm for nurse scheduling problem showed that the incorporation of local search at intialization phase can create better individuals for selection purposes. In job shop scheduling problem, Qing-dao-er-ji and Wang (2012) made a comparison between the algorithm that incorporated Local Search operator and without Local Search operator. They showed either equal or better improvement in the solution. A research conducted in menu planning problem by Mohd Razali (2011) provided a feasible solution for all cases when she hybridized Local Search operator in initial phase and final phase of GA. The mechanism also acted as repair operator when infesible solution was obtained.

### 3.7 Summary

This chapter has explained the fundamental concept of Evolutionary Algorithm and the operators involved. Then, the techniques for handing constraints are discussed. Previous related studies in EA have also been presented. The following chapter will discuss the whole research methodology beginning from problem definition until the verification of the result obtained.

# CHAPTER 4

# RESEARCH METHODOLOGY

The main problem of aquaculture diet formulation, especially in shrimp production, has been identified in previous chapters. We have also identified the promising technique that can be utilized to solve the diet formulation problem. Consequently, this chapter describes the methodology to achieve the objectives. Overall, the research activities designed to accomplish the objectives are illustrated in Figure 4.1. Next, Figure 4.2 details every phase along with the techniques employed to accomplish the particular objective.

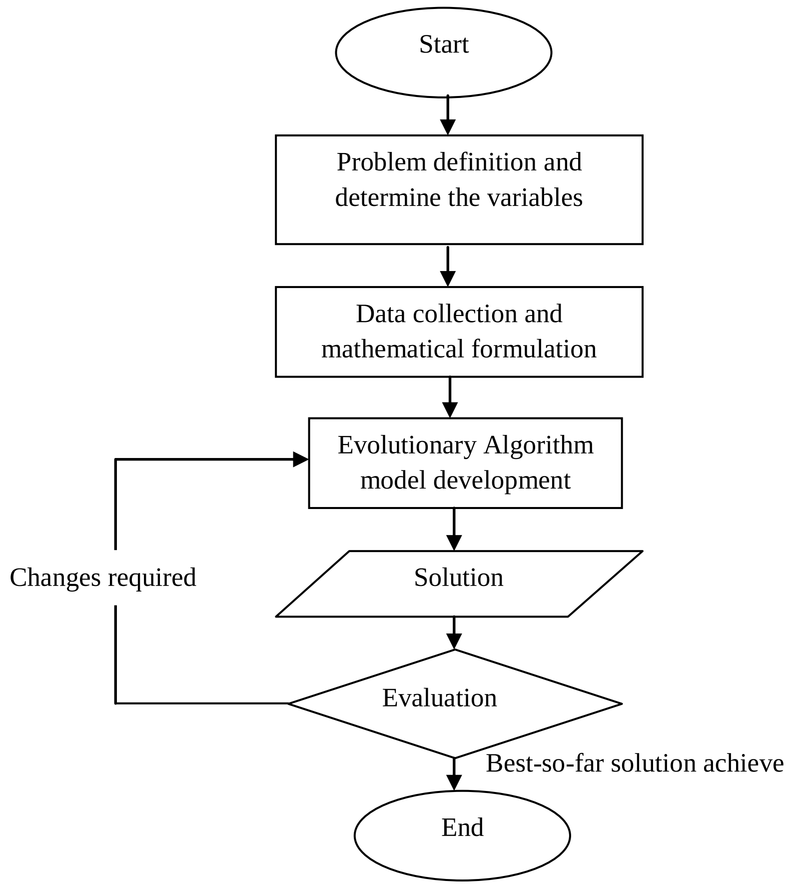
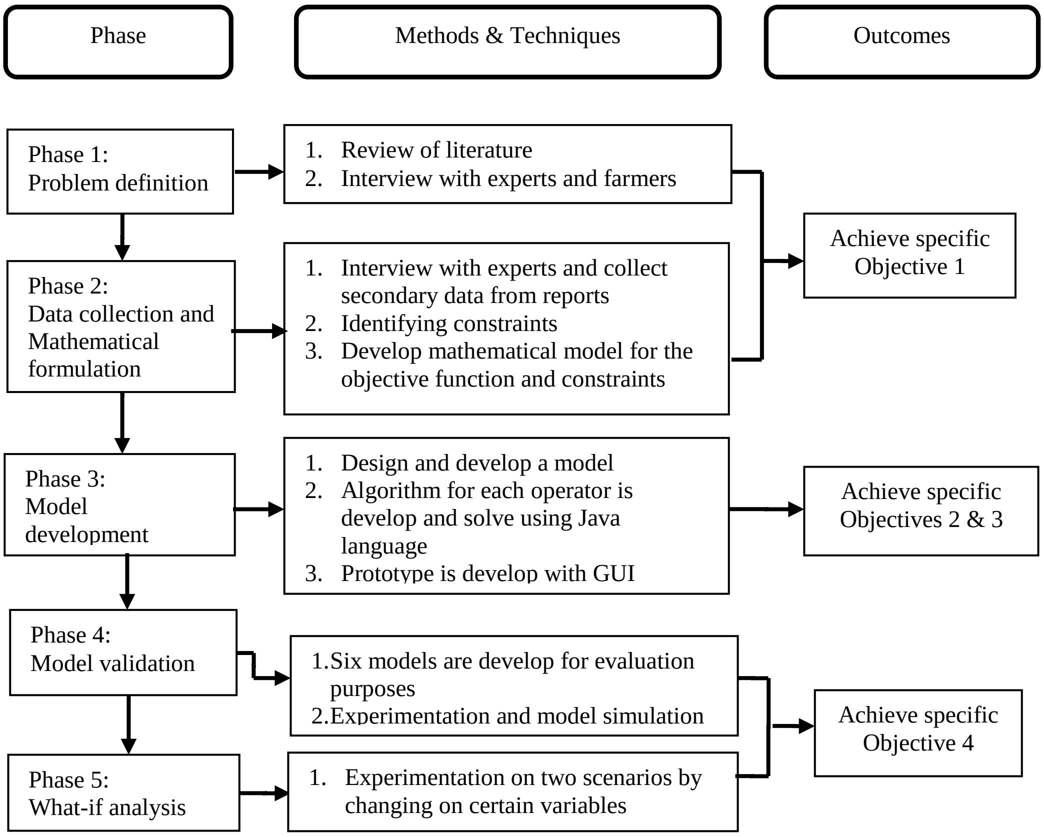



*Figure 4.1.* Flow of Research Activities




**OBJECTIVE**

The primary objective of this thesis is to develop a model that can lead to the creation of shrimp feed mix that will be able to meet the nutritional requirements for effective production. In order to reach the goal, some specific objectives need to be fulfilled. These are as follows:

1. To identify the maximum and minimum requirements of shrimp feed for various aspects.
2. To construct a new filtering heuristics known as Power Heuristics as part of the initialization procedure that is capable of filtering some combinations of ingredients from a selected database of choices, which could lead to potentially poor solution.
3. To construct a new crossover operator known as Average Crossover that is able to produce potentially good solution.
4. To conduct a comparative evaluation on the solutions based on several evolutionary models generated and what-if analyses.

*Figure 4.2.* Details of Research Activities

### 4.1 Research design

This research aims to develop a model for shrimp diet that can satisfy the nutritional requirements of shrimp with minimum cost. To achieve the objective, this research is conducted through five phases of research activities; problem definition, data collection, model development, and model evaluation. In model evaluation, model validation and what-if analysis are involved (refer to Figure 4.1 and 4.2). Data collection involves primary and secondary data. Primary data are collected through interviews with some experts in shrimp and aquaculture field, while secondary data are retrieved from reports by National Research Council (NRC), literature review, and website. After identifying all the constraints, the diet formulation is written in a mathematical formulation form. Three variants of the model and three sub-models of each variant are then developed using enhanced Evolutionary Algorithm (EA) with new proposed selection and crossover operators. The proposed model is then evaluated through validation and what-if-analysis. Experimentation and model simulation is employed to validate the model. Meanwhile, in what-if analysis, two scenarios are experimented when changes to total ingredient weight and price of ingredients are made.

#### 4.1.1 Problem definition

The main concern of this research is to develop a diet formulation model for juvenile Whiteleg shrimp that satisfies the required constraints at minimum cost. Whiteleg shrimp is chosen because this species is the most popularly cultured shrimp in Asia and Malaysia. Its production contributes to nearly 80% of total shrimp production in Malaysia (FAO, 2013).

The problem of this shrimp production consists of several constraints, which include total ingredient weight, nutrient range, and ingredient range, defined based on experts' opinion and literature review. Nutrient range is classified into three types which are single nutrient, combination of nutrients, and ratio between two nutrients. In this problem, a list of selected ingredients in specific quantities that satisfy the required constraints at the end of the problem shrimp process is identified. The ingredient mix represents the diet formulation for the shrimp.

### 4.1.2 Data collection and mathematical formulation

Data collection is one of the most important components of research since data are needed to verify whether the proposed model can be accepted or not. In this research, both primary data and secondary data were collected. Primary data were collected through interviews with experts. Two of the experts are researchers from the National Prawn Fry Production and Research Centre (NAPFRE), which is under the Department of Fisheries Malaysia. NAPFRE is a research centre specializing in shrimp that provides shrimp fry and training for farmers. One of the experts is an academician from the Aquaculture Department in Universiti Putra Malaysia (UPM), whose area of expertise is aquaculture diet. In addition to these experts, interviews with commercial manufacturers of shrimp feed and farmers were also conducted to understand the current scenario of shrimp feed in Malaysia. Meanwhile, secondary data were taken from two reports published by the National Research Council (NRC). The first report titled *United States-Canadian Table of Feed Composition* (NRC, 1982) and the second report is *Nutrient Requirement of Fish and Shrimp* (NRC, 2011). Some of the secondary data were taken from literature and website proposed by the experts. The details of data collection can be referred in Section 4.2.

Model formulation is the process of writing mathematically the specific problem in order to solve the diet formulation problem through the proposed technique. Model formulation consists of an objective function and a list of constraints to satisfy. The objective function for this diet formulation problem is to minimize the penalty value subject to constraints such as total ingredient weight, ingredients' restriction, and nutrient restrictions. Section 4.3.5 and 4.3.6 discusses in detail the model formulation developed.

### 4.1.3 Model development

Diet formulation model was developed using the technique of Evolutionary Algorithm (EA) with the enhancement of some EA operators. All three model variants with different selection operators and three sub-models of each variant with different crossover operators were developed for evaluation purposes. The algorithms for each model were then develop and solved using Java language. System prototype was also developed using Java language. Graphical User Interface (GUI) was designed to ease the communication process between user and the proposed model. The user can choose fourteen preferred ingredients among ninety-one available ingredients in the database to be counted into the calculation. Besides, total ingredient weight might be determined by the user in order to satisfy the user needs without under or overestimating the bulkiness of the ration. Detailed explanation about the model development can be found in Section 4.4.

### 4.1.4 Model validation

Experimentation on each stage in the EA operators, which are initialization, parent selection and crossover, was made to verify that the operators are valid. The experimentation on parameter values was also required to test the best value to be applied to our model. Additionally, several comparisons to measure variations of the model were made in order to confirm that the proposed model could produce acceptable good solutions. All experimentations are discussed in more detail in Chapter 6.

### 4.1.5 What-if analysis

What-if analysis was implemented to test the operability of the proposed model in different scenarios. It is eventually known as sensitivity analysis, as behavioural performance of the model was observed when changes were made. Two scenarios were tested and the result of each solution was used as a measurement. The first scenario was by increasing the total ingredient weight and the second was by decreasing the price of each ingredient. The cost of total feed and the effect on the feasible solution was then observed. The implementation details of this analysis are shown in Section 6.6.

## 4.2 Types of data and data collection

Two types of data, primary data and secondary data, were collected for this research. In order to be familiar with aquaculture and shrimp environment, interviews with a group of experts and shrimp feed manufacturers were conducted. In addition to primary data gathered from the interview sessions, secondary data consisting of different types of dataset were also required to evaluate the proposed algorithm. Thus, several types of data needed in this research were identified along with their sources are explained in detail in the next subsection.

### 4.2.1 List of nutrients in acceptable range

Sixteen nutrients were considered in this research such as crude protein, lipid, fibre, ash, calcium, phosphorus, and ten EAA. Besides, two nutrient combinations and one ratio between two nutrients were also considered. A list of all nutrients and its range required by juvenile Whiteleg Shrimp was obtained from several sources as depicted in Table 4.1. Some of the nutrients required by Whiteleg shrimp were not yet determined. But based on expert opinion, the requirement value is located based on the general shrimp feed requirement.

- Nutrient's priority

Based on expert opinion, the nutrients are formulated based on its priority. The list of priorities is as below:

1. Approximate: consists of crude protein, lipid, ash, fibre, calcium and phosphorus
2. Amino acid: consist of ten EAA
3. Fatty Acid
4. Omega
5. Combination of all above

In this research, the combination of first and second was taken into account, while the third and fourth nutrients will be considered in our future research.

Table 4.1

*List of Nutrients within Its Range*

| Nutrients | Minimum | Maximum |
|---|---|---|
| *Single nutrient* | | |
| ***Approximate*** | | |
| Crude Protein, % | 38.00[a] | 45.00[c] |
| Lipid, % | 0.08[b] | 0.18[b] |
| Fibre, % | - | 4.00[a] |
| Ash, % | - | 15.00[a] |
| Calcium, % | - | 2.30[a] |
| Phosphorus, % | 0.30[b] | 0.70[b] |
| ***EAA*** | | |
| Arginine, % | 1.60[b] | 2.32[a] |
| Histidine, % | 0.60[b] | 0.84[a] |
| Isoleucine, % | 1.00[b] | 1.33[a] |
| Leucine, % | 1.70[b] | 2.16[a] |
| Lysine, % | 1.55[b] | 1.65[b] |
| Methionine, % | 0.70[b] | 0.96[b] |
| Phenylalanine, % | 1.40[b] | 1.60[b] |
| Threonine, % | 1.30[b] | 1.44[a] |
| Tryptophan, % | 0.20[b] | 0.32[b] |
| Valine, % | 1.20[b] | 1.60[a] |
| *Nutrients combination* | | |
| Methionine + Cystine, % | 1.00[b] | 1.44[a] |
| Phenylalanine + Tyrosine, % | 2.70[a] | 7.10[c] |
| *Nutrients ratio* | | |
| Calcium: Phosphorus, % | 1:1.3[c] | 1:1.3[c] |

[a]: Akiyama (1992)
[b]: NRC (2011)
[c]: expert opinion

### 4.2.2 Ingredients and price

Hundreds of ingredients have been explored in literature of nutrition to be used as marine feed including shrimp. However, the ingredients' restriction (minimum and maximum) is hard to get from previous studies since normally they only proposed a new suitable ingredient. However, based on expert opinion, feed ingredient originally from animal must be at least 10% included in the formula and no maximum value is applied. On the other hand, for plant based ingredient none of the ingredient can be considered as the minimum value and not more than 40% is allowed. Basically, the more animal based ingredient is put in the formulation, the better combination of ingredient can be obtained. Unfortunately, since the price of animal based is too expensive, substitution of plant based ingredient is then measured. A list of suitable ingredients obtained from experts as well as from past studies in this area can be viewed in Appendix B. The price of the ingredients is quoted in USD. The currency rate on October 7, 2012 was RM 3.05350 for 1 USD.

### 4.2.3 Nutrient composition in each ingredient

Nutrient composition in each ingredient was taken from the nutritional requirement report published by NRC (1982) and NRC (2011) and from expert's opinion. A list of complete nutrient composition of feed ingredients for shrimp is attached in Appendix C.

## 4.3 The constraints of diet formulation model

The main intention of this model is to minimize the shrimp feed cost and at the same time fulfil the entire shrimp requirement. Therefore, before developing the model, the constraints in shrimp feed should be further investigated. There are two categories of constraints that exist. They are ingredient constraint and nutrient constraints. Ingredient constraints consist of ingredients range, total ingredient weight, and maximum number of ingredients. On the other hand, nutrient constraints consist of single nutrient, combination, and ratio of nutrients. In this problem, hard constraints are maximum number of ingredients, total ration weight and a single nutrient that is crude protein. Other constraints are soft constraints that can be violated but with certain penalty values. Mitani and Nakayama (1997), Şahman et al.

(2009) and Zhang and Roush (2001), are among others who considered these contraints as soft constraints.

### 4.3.1 Ingredients range constraints

Ingredients range constraint is actually based on previous work and data were collected from experts' opinion. All the ingredients have their minimum and maximum restriction value as lower bound and upper bound respectively in order to produce palatable diet (NRC, 2011). The combination of ingredients was used to design the pallet which gave the texture and taste of the pallet. For example, too much maize in the pallet will make the pallet yellowish. This problem is not harmful the shrimp but might affect their appetite. Therefore, if the ingredient(s) obtained by the model are not equal to the ingredients restrictions, a penalty value is given for each ingredient. The penalty value varies for each ingredient based on expert opinion that is based on digestibility of nutrients. A list of penalty value of ingredients can be seen in Table 5.1 in Section 5.2.2.1.

The data of ingredients range consisting of minimum and maximum requirement of ingredients can be viewed in Appendix B. The listed ingredients can be put either within the minimum and maximum percentage or they do not have to be considered at all.

### 4.3.2 Total ingredients' weight constraints

Weight of ingredient is another constraint considered in this research. The user is allowed to put any weight from 100 kg up to 10,000 kg or 10 tonne at interface. If the weight obtained by the system is not equal to the weight determined by the user, a variety of penalty value is given to the solution in five ranges:

1. within 0.5 kg from total ingredient weight

   No penalty is given

2. within 2 kg from total ingredient weight

   100 penalties is given

3. within 5 kg from total ingredient weight

   200 penalties is given

4. within 10 kg from total ingredient weight

   400 penalties is given

5. more than 10 kg from total ingredient weight

   1500 penalties is given

These penalty values are given based on the importance of total ingredient weight to the prototype user such as farmer and manufacturer to obtain a specific weight. In this case, we accept maximum difference until 10kg with a big penalty value. The solution with more than 10kg difference is rejected. This constraint is design based on range to allow some relaxation since some violations do not affect shrimp health but affect economically the diet, as concluded by Lara and Romero (1992), and Mohr (1972).

### 4.3.3 Maximum number of ingredients constraint

Based on expert opinion, the total number of ingredient in a mix is at most ten in order to save on transportation cost especially for imported ingredients. However, based on literature and commercial data collected from feed factory, at most fourteen ingredients were employed in our model. In fact, the more combinations of ingredient, the greater quality of feed is obtained.

### 4.3.4 Nutrients range constraints

Similar to ingredients constraint, nutrient requirements must be within the range of minimum and maximum value. But this constraint will give more impact to the shrimp compare to ingredient constraint. Over or under formulated of nutrients, might delay the shrimp growth and prolong the breeding time in ponds. As a result, the production cost will be increased. In this research, nutrients' range is divided into three categories which are single, combination, and ratio.

#### 4.3.4.1 Single nutrient

Single nutrient consists of standalone nutrient available in the system including crude protein, lipid, fibre, ash, calcium, phosphorus, and 10 types of EAA. The following table depicts the range of single nutrient requirement for shrimp.

Table 4.2

*Nutrients Range for Single Nutrient*

| **Nutrients** | **Minimum** | **Maximum** |
|---|---|---|
| Crude Protein, % | 38.00 | 45.00 |
| Lipid, % | 0.08 | 0.18 |
| Fibre, % | 0 | 4.00 |
| Ash, % | 0 | 15.0 |
| Calcium, % | 0 | 2.30 |
| Phosphorus, % | 0.30 | 0.70 |
| Arginine, % | 2.2 | 2.32 |
| Histidine, % | 0.6 | 0.84 |
| Isoleucine, % | 1.0 | 1.33 |
| Leucine, % | 1.7 | 2.16 |
| Lysine, % | 1.55 | 1.65 |
| Methionine, % | 0.7 | 0.96 |
| Phenylalanine, % | 1.4 | 1.6 |
| Threonine, % | 1.3 | 1.44 |
| Tryptophan, % | 0.2 | 0.32 |
| Valine, % | 1.4 | 1.6 |

A penalty value of 1500 is given for crude protein if the value achieved from the model violates the range of nutrient as given in Table 4.2. Crude protein is considered as hard constraint because it is the most important nutrient needed in the animal body and the formulation of this nutrient is calculated at the early introduction of diet formulation problem. For other approximate nutrients that are lipid, fibre, ash, calcium and phosphorus, a penalty value of 40 is given if the nutrients are violated. Else, for amino acid, a penalty value of 30 is given. This strategy is adopted from other researchers (Şahman et al., 2009; Tozer & Stokes, 2001), who considered all nutrients range constraint as soft constraints.

#### 4.3.4.2 Combination of nutrients

Two nutrient combinations are considered in this research as depicted in Table 4.3. Thus, if the nutrient combination achieved from the system violates the stated value, a penalty value of 20 is given for each nutrient combination.

Table 4.3

*Nutrients Range for Combination of Nutrient*

| Nutrients | Minimum | Maximum |
|---|---|---|
| Methionine + Cystine, % | 1.0 | 1.44 |
| Phenylalanine + Tyrosine, % | 2.7 | 7.1 |

#### 4.3.4.3 Ratio of nutrient

Nutritional imbalance is one of the issues in diet formulation. Nutritional imbalance occurs if the nutritional value is over or under formulated. Beginning in the year 2000, nutritional issues become more global. The nutrient ratio among certain nutrients is identified as one of the sources that contribute to this issue. Thus, in this study, we considered the ratio of nutrient as one way to obtain a balanced nutrient value. In shrimp feed, our focus is on the ratio of calcium to phosphorus, as 1:1.3. The constraint is considered as soft constraint; thus a penalty value of 20 is given for each constraint violated.

### 4.3.5 Objective function of diet formulation model

The aim of this study is to fulfil nutritional requirements with minimum cost subject to certain constraints. The objective function is to minimize the penalty function value while minimizing cost.

The cumulative feed cost is defined as summation of the weight for all ingredient times the cost in one kg for each ingredient.

$$f(s) = \min \sum_{i=1}^{n} (X_i \times C_i) \tag{4.1}$$

where $C_i$ is the cost of ingredient $i$ in one kilogram,

$X_i$ is the weight of the i*th* ingredient,

*s* is the cumulative cost in a string of chromosome

and *n* is the number of ingredient

### 4.3.6 Mathematical formulation

The constraints of the diet formulation model consist of ingredients range, ingredient weight, maximum number of ingredients, and nutrients range.

i. Ingredients range constraint

Ingredients range should be equal to zero or within the minimum and maximum requirement of each ingredient. Minimum and maximum requirement is different on each ingredient (refer to Appendix B).

$X_i = 0$ or $L_{X_i} \le X_i \le U_{X_i}$ for all $X_i$ (4.2)

$L_{X_i}$ = lower bound of ingredient $i$

$U_{X_i}$ = upper bound of ingredient $i$

$X_i$ = the weight of the i*th* ingredient

ii. Ingredients' weight constraint

The summation of all selected ingredients should be equal to the weight predefined by user in the GUI.

$$\sum_{i=1}^{n} X_i = Y \tag{4.3}$$

$Y$ is a weight predefine by user in user interface

iii. Maximum number of ingredient constraint

Total number of selected ingredients should be at most 14.

$$n \le 14 \tag{4.4}$$

iv. Nutrients range constraint

This constraint is divided into three categories: single nutrient range, combination of nutrient, and ratio of nutrient.

- Single nutrient

The general model for a single nutrients range is total nutrient $k$ in the final ration should be within the permitted range of that nutrient.

$$L_{N_k} \le \sum_{i=1}^{n} N_{ki} X_i \le U_{N_k} \tag{4.5}$$

$N_{ki}$ = Total value of nutrient $k$ in ingredient $i$, $k$=1,2,…,16

$L_{Nk}$ = lower bound of total value of nutrient $k$

$U_{Nk}$ = upper bound of total value of nutrient $k$

- Combination of nutrient

Two nutrient combinations are considered in this study that is the combination of methionine and cysteine, and the combination of phenylalanine and tyrosine.

Methionine+Cystine:

Total combination of methionine and cystine should be within the allowable range of this combination.

$$L_{(Met+Cys)} \leq \sum_{i=1}^{n} (Met + Cys)_i X_i \leq U_{(Met+Cys)} \tag{4.6}$$

Phenylalanine+Tyrosine:

Total combination of phenylalanine and tyrosine should be within the allowable range of this combination.

$$L_{(Phe+Tyr)} \leq \sum_{i=1}^{n} (Phe + Tyr)_i X_i \leq U_{(Phe+Tyr)} \tag{4.7}$$

$(Met+Cys)_i$ = Combination of Methionine and Cystine content in ingredient $i$

$(Phe+Tyr)_i$ = Combination of Phenylalanine and Tyrosine content in ingredient $i$

- Ratio of nutrient

Calcium: Phosphorus:

The ratio of calcium to phosphorus should be within the allowable range.

$$L_{ratio} \leq \frac{\sum_{i=1}^{n} Ca_i}{\sum_{i=1}^{n} Pho_i} \leq U_{ratio} \tag{4.8}$$

$L_{ratio}$ = lower bound of ratio between calcium to phosphorus in ingredient $i$

$U_{ratio}$ = upper bound of ratio between calcium to phosphorus in ingredient $i$

## 4.4 Adaptation of diet formulation problem into Evolutionary Algorithm

After the model is formulated, the modelling process begins by adapting the parameters into the EA. The proposed EA model is shown in Figure 4.3.

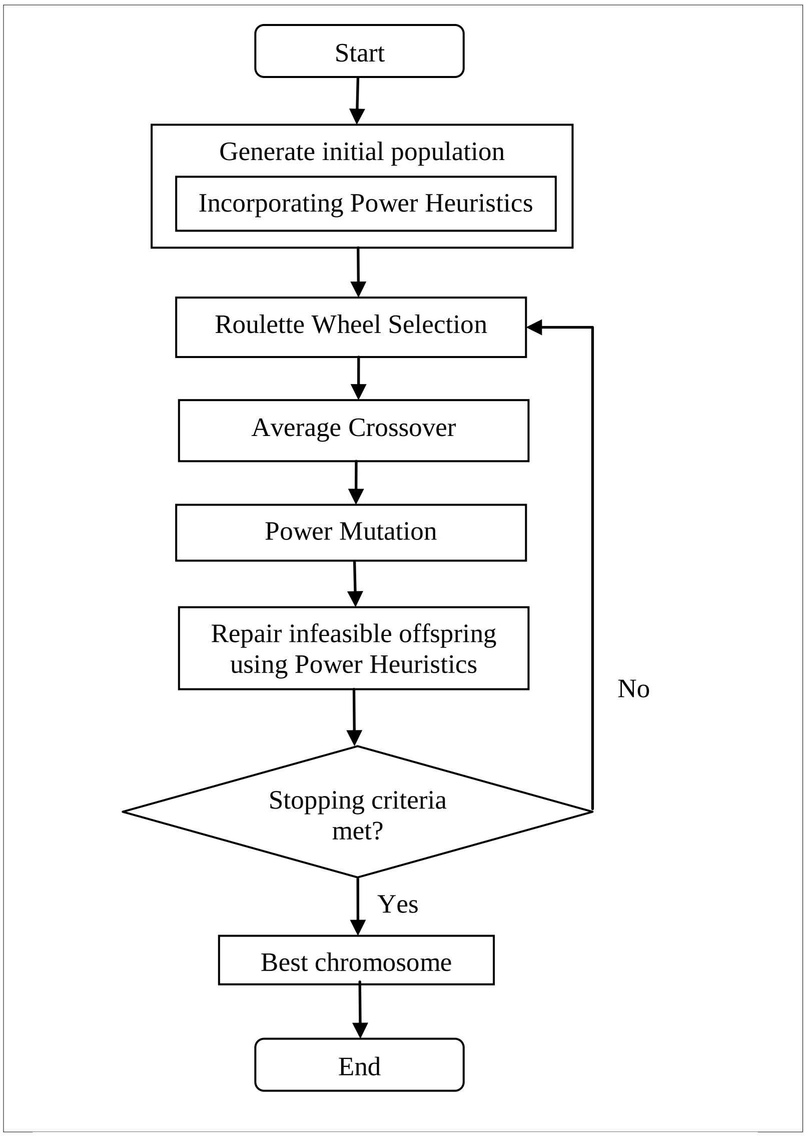



*Figure 4.3.* EA-PH-RWS-Avg Model

Power Heuristics is inserted in both initialization and mutation operators. The purposes of Power Heuristics are to filter the unwanted combinations of ingredients and search for a lower penalty value. The proposed EA-PH-RWS-Avg Model employs Roulette Wheel Selection and Average Crossover operators. Average Crossover is a new version of crossover modified from the traditional One-Point Crossover. This operator is introduced in this model to evaluate the performance of the proposed crossover with the established One-Point Crossover.

In order to evaluate the proposed model, two selection techniques were designed, which are the Queen-Bee Selection and Roulette-Tournament Selection. Therefore, two models with each selection operator are developed. In addition, two sub-models with different crossover operators are also developed for each model for comparison purposes. Figure 4.3 until Figure 4.8 represents all models involved in this research. Each model requires the same starting step, that is the initialization incorporating the Power Heuristics. After that, selection and crossover take place. The process is then continued with Power Mutation and Power Heuristics, consecutively. The whole process is repeated until a number of generations is reached.

This section continues with presentations shows the differences in each model through from the following figures. Then, a detailed description of each new EA operator is explained after each figure.

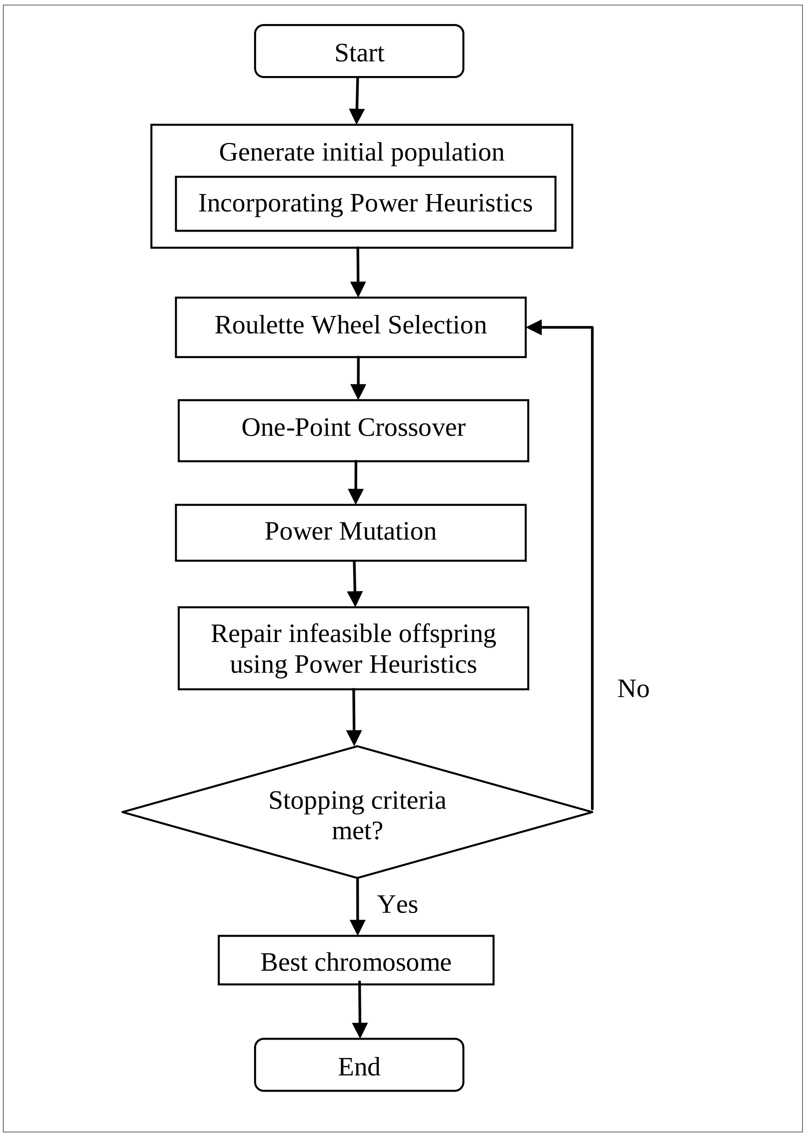



*Figure 4.4.* EA-PH-RWS-One-Pt Model

EA-PH-RWS-One-Pt Model employs the traditional Roulette Wheel Selection and One-Point Crossover. In this model, there is no introduction to any new procedures except for the Power Heuristics. Power Heuristics acts as a repair operator in the initialization and mutation stages.

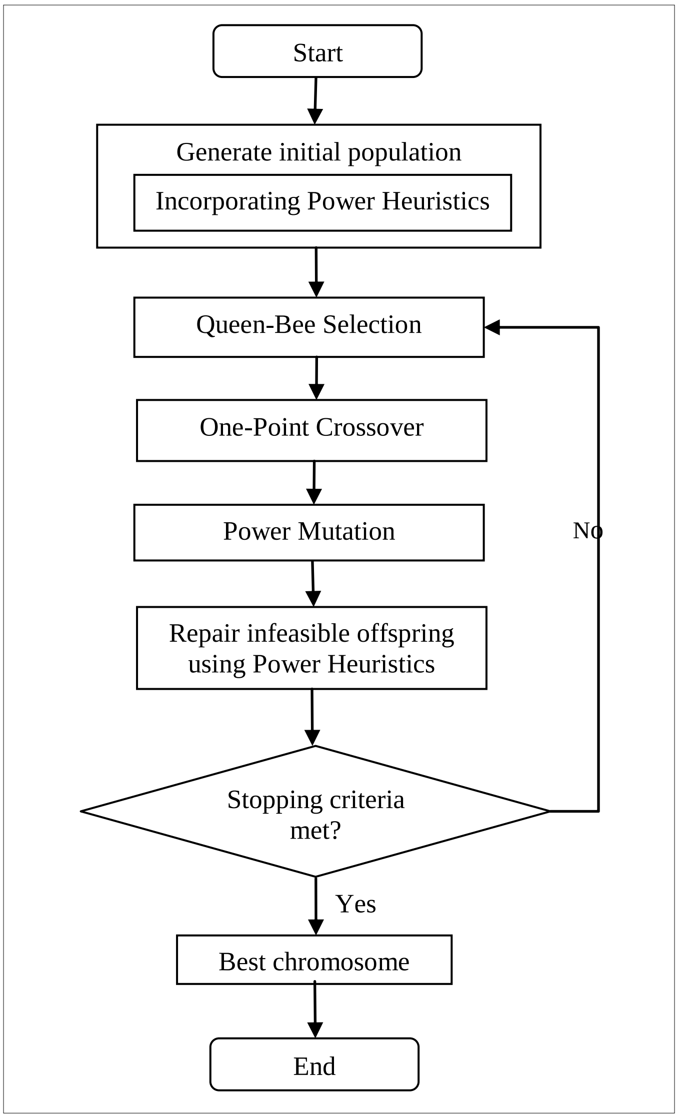



*Figure 4.5.* EA-PH-QB-One-Pt Model

In this model, Queen-Bee Selection is employed that uses the fittest offspring in each generation. Queen-Bee is adopted from queen bees' behaviour (Karaboga, 2005) that has the most powerful authority in a bee swarm. The standard One-Point Crossover is adopted in this model.

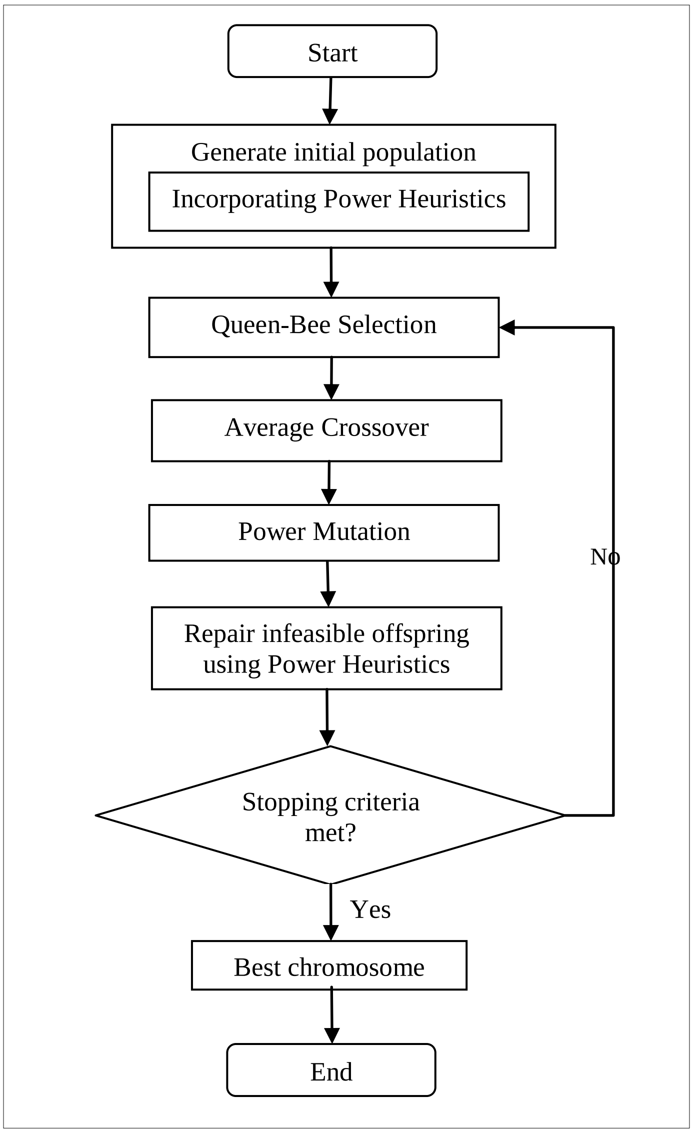



*Figure 4.6.* EA-PH-QB-Avg Model

EA-PH-QB-Avg is a sub-model of EA-PH-QB-One-Pt Model. The Queen-Bee Selection with a new Average Crossover is adopted in this model. The combination of Average Crossover with Queen-Bee Crossover is evaluated to record the model performance.

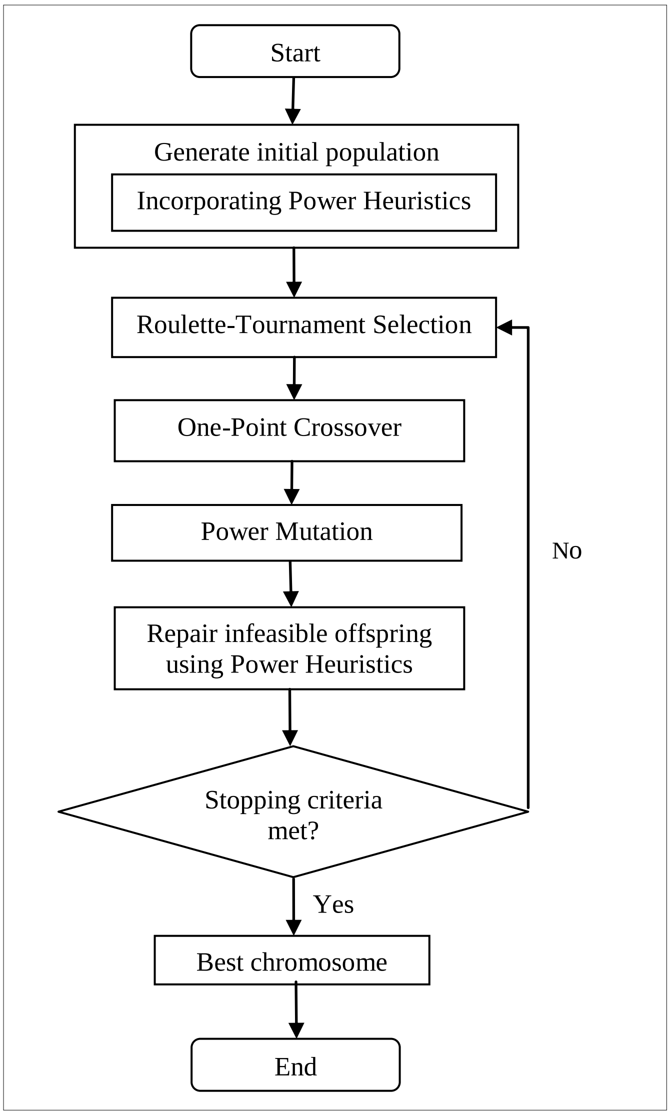



*Figure 4.7.* EA-PH-RT-One-Pt Model

Roulette-Tournament Selection is a new selection operator introduced in this model. It is a combination of an established feature of Roulette Wheel Selection and the Tournament Selection.

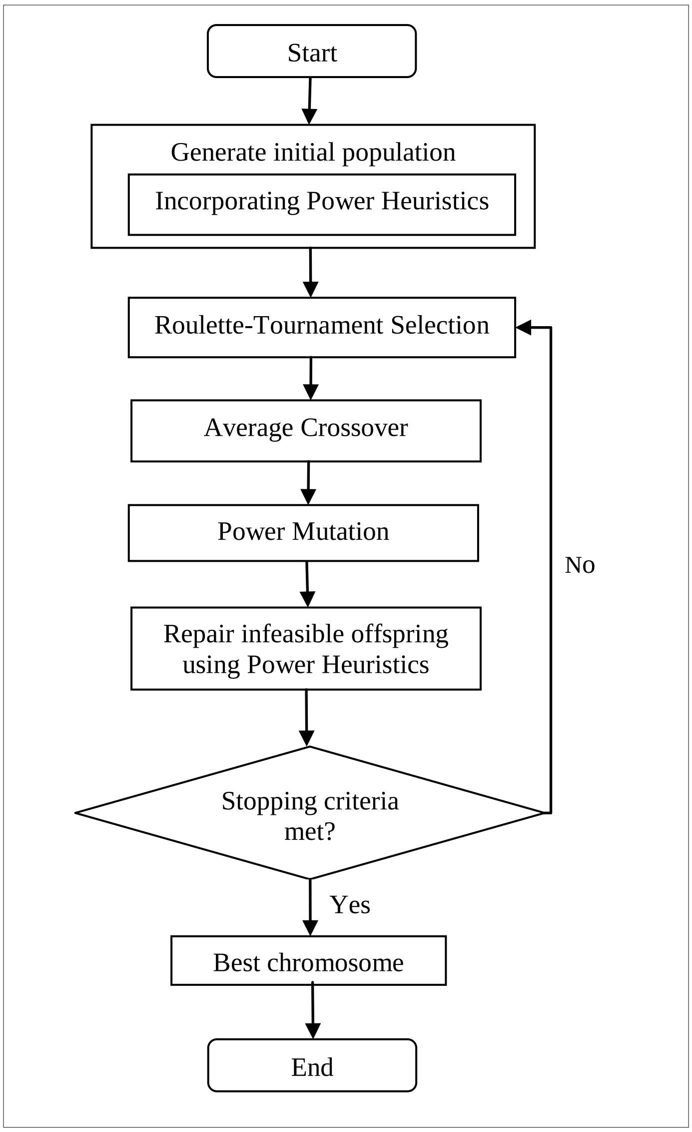



*Figure 4.8.* EA-PH-RT-Avg Model

EA-PH-RT-Avg is a sub-model of EA-PH-RT-One-Pt Model. This model incorporates a new mechanism for selection by introducing the Roulette-Tournament Selection, and also for crossover by the Average Crossover operator.

### 4.4.1 Representation structure

The solution is represented in real value encoded with an array of 1x14 values. The number of ingredients corresponds to the number of values. In this research, the representation of solution is as shown in Figure 4.9. The value inside the cell represents the ingredient's weight (in kg) which can take a decimal point value.

Ingredients, $X_i$

| $X_1$ | $X_2$ | $X_3$ | $X_4$ | $X_5$ | $X_6$ | $X_7$ | $X_8$ | $X_9$ | $X_{10}$ | $X_{11}$ | $X_{12}$ | $X_{13}$ | $X_{14}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 6.1 | 28 | 3 | 40 | 37 | 6 | 8 | 3 | 20 | 8 | 9 | 3.5 | 4.6 | 5 |

$i$ = 1, 2, .......14

*Figure 4.9.* Representation Structure

### 4.4.2 Solution initialization

The initial population is generated in semi random since the upper limit and lower limit of the ingredient is considered as the limitation while generating a solution. After the solution has been initialized, the penalty value is calculated. While calculating the penalty value, the system is checked for the solution whether it is feasible or not.

#### 4.4.2.1 Penalty computation

The feasibility of a chromosome depends on constraints violation. The constraints are divided into two types: hard constraints and soft constraint. If hard constraint is violated, the solution is not feasible. Soft constraint can be violated; but the more minimal penalty, the better solution is generated. The list of constraints can be referred to in Section 4.3. If the solution is infeasible, it means that the hard constraint is violated and 1500 penalties are imposed as a punishment for each constraint violated.

#### 4.4.2.2 Power Heuristics operator

A special Power Heuristics acts as a filtering operator and is incorporated in this initial operation with the intention to reduce the number of ingredients or slightly change the ingredient value to meet the constraints requirement such as total weight and nutrient range. This step applies mutation mechanism in order to modify solution to produce a feasible solution or at least reduce the penalty value. The original idea of this operator comes from Power Mutation mechanism as introduced by Deep and Thakur (2007). Power Heuristics uses local searching concept that searches around the neighborhood solution.

Power Heuristics begins by checking the solution whether it is feasible or not. If a penalty value exists, it means that the solution is infeasible. Then, a new value of ingredient will be searched within the range of one from the current value of the ingredient. This mechanism can remove a few ingredients based on random number generated. Finally, the penalty value for the new solution is calculated.

### 4.4.3 Parents selection operator

Roulette Wheel Selection (RWS) is a classic selection operator in the proportional type. In this study, RWS is adopted. Previous researches have shown that a good result can be obtained by using RWS operator in hybrid GA (Duan, Xu, & Xing, 2010; Mohd Razali, 2011). However, due to the disadvantages of RWS for minimization problem, first, a modification is made to transform this problem.

In addition to RWS, another selection operator carried out is Queen-Bee Selection. This selection operator chooses the fittest chromosome as parent one and then randomly chooses another chromosome as parent two. The idea of Queen-Bee Selection is injection of bee's characteristic into the selection operator. The incorporation of Queen-Bee Selection operator has been done since 2003 by Jung. The idea was then modified by Azeem and Saad (2004). Other versions of Queen-Bee operator were also introduced by Karci (2004), Xu et al., (2008), and Lu and Zhou (2008). The basic idea in these researches is that the least total penalty in the population is chosen as a queen bee, which is then crossed over with any drone as its couple. Queen bee concept can increase the exploitation in EA but somehow increases the opportunity fall into premature convergence (Jung, 2003; Xu et al., 2008). This is caused by the selection mechanism that takes only the fittest chromosome and eliminates the less fit chromosome (Sivaraj & Ravichandran, 2011).

The third selection operator introduced in this study is the combination of RWS operator and Binary Tournament Selection. The operator starts with the same steps as RWS. Then, the Binary Tournament operator takes place to choose two chromosomes as parents. Like Binary Tournament, two chromosomes are randomly picked from all solutions, and the fitter parents will be chosen as parent one. The same step is repeated to find parent two. The hybridization of this operator will merge the advantages from both RWS and Binary Tournament.

### 4.4.4 Crossover operator

As mentioned in Section 3.4.4, various types of crossover have been identified in the literature. In this research, the classical One-Point Crossover is adopted with midpoint selected as the cutting point. Multi-point is not employed in this research since the structure of genes is short, that is fourteen. Therefore the chances of a population become homogenous after many generations as highlighted by Spears and De Jong (1990) is high. Besides, a new type of crossover named Average Crossover is also created in this research. This new idea of crossover came from trying to find variation in crossover in queen bee concept by Jung (2003). The motivation for applying Average Crossover is to observe its potential with other established crossover operators. This type of crossover is basically similar with traditional One-Point Crossover except that it has extra computation to find a midpoint between two parents. Figure 4.10 illustrates the mechanism of Average Crossover in more detail.

Parent 1:

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| a | b | c | d | e | f | g | h |

Parent 2:

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| k | l | m | n | o | p | q | r |

Child 1:

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| a | b | c | d | (e+o)/2 | (f+p)/2 | (g+q)/2 | (h+r)/2 |

Child 2:

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| (a+k)/2 | (b+l)/2 | (c+m)/2 | (d+n)/2 | e | f | g | h |

*Figure 4.10.* Procedure of Average Crossover

### 4.4.5 Mutation operator

In simple GA (Goldberg, 1989), one mutation technique is employed in the mutation step. In our study, Power Mutation is adopted followed by Power Heuristics to once again filter some ingredients to find a better solution with less penalty value.

#### 4.4.5.1 Power Mutation

In this research, Power Mutation introduced by Deep and Thakur (2007) is chosen among other mutation operators. This operator is adopted since it is introduced purposely for real number representation structure. The advantage of Power Mutation is that it takes into consideration on the boundary of constraint and thus possibility to violate the constraint is reduced.

#### 4.4.5.2 Power Heuristics operator

Power Heuristics undertakes a repair operator which modifies the solution if it is still infeasible. However, if a feasible solution still cannot be obtained, this step can reduce the penalty function value. The most important task of Power Heuristics is to filter some ingredients and remove them from calculation. Power Heuristics is very useful in situations of many ingredients that can be considered as shrimp diet. This operator is capable of filtering some combinations of ingredients from a selected database of choices, which could lead to potentially poor solution.

### 4.4.6 Stopping criterion

A number of generations are chosen to be the stopping criterion for this system. This strategy is adopted from Şahman et al. (2009). After the loop is completed, the best-so-far solution is presented as a list.

### 4.4.7 Diet formulation prototype

A diet formulation prototype is developed for users including nutritionists, manufacturers, and farmers with the intention to obtain a healthy diet. GUI is developed for the users to choose the preferred ingredients and define the required ingredient weight. The final solution appears at the interface that shows the best combination of ingredients that satisfies the constraints at a minimum cost.

### 4.5 Model validation

A few comparisons are made to validate the performance of the proposed algorithm. A comparison between One-Point Crossover and Average Crossover is made to observe the performance of the proposed Average Crossover. Another comparison is conducted between three selection mechanisms: Roulette Wheel, Queen-Bee, and Roulette-Tournament.

### 4.6 What-if analysis

Changing the dataset in some variables is done in what-if analysis to evaluate the proposed algorithm to adapt with different scenarios. This experiment is required to observe the flexibility of the proposed system prototype.

#### 4.6.1 Experiment of increase total ingredient weight

In model validation, the total ingredient weight is set to 100 kg. Therefore, in this experiment, the total ingredient weight is increased to 500 kg. The solution is observed especially for the total penalty value and processing time.

#### 4.6.2 Experiment of decrease ingredient price

This experiment is done by cutting the price of all fourteen ingredients involved in the study. The price of each ingredient is decreased by RM 0.30. The impact of the solution is then observed.

## 4.7 Summary

The proposed methodology focuses on the diet formulation problem using Evolutionary Algorithm approach to satisfy the constraints at a minimum cost. The incorporation of Power Heuristics adds value of this method as it can filter many ingredients at a time in order to find the feasible combination of ingredients. Moreover, the Power Heuristics also acts as filter operator that controls the quality of chromosome going into the population. As a result, a better chromosome will be selected as parents in the next stage. These chromosomes obtained from Power Heuristics may not feasible but may having a less penalty value.

In summary, this chapter has answered the first specific objective mentioned in the initial chapter. The objective is to identify the maximum and minimum requirements of shrimp feed for various aspects. The next chapter discusses how the second and third specific objectives are to be met. The second specific objective pertains to constructing a new filtering heuristics known as Power Heuristics as part of the initialization operation that is capable of filtering some combinations of ingredients from a selected database of choices, which could lead to potentially poor solution. The third specific objective is to construct a new crossover operator known as Average Crossover that is able to produce a potentially good solution.

# CHAPTER 5

# THE DEVELOPMENT OF DIET FORMULATION PROTOTYPE

This chapter describes the steps involved in the development of a diet formulation model and its implementation through our proposed Evolutionary Algorithm. Firstly, in Section 5.1, a brief description of shrimp diet formulation is explained to recollect the problem definition. A detailed procedure of the proposed algorithm is discussed in Section 5.2, followed by the implementation of a diet formulation prototype in Section 5.3. The approach to achieve our second and third specific objectives is related to the development of our proposed algorithm, discussed in Section 5.2. A simple graphical user interface is developed to ease the communication process between users and system for implementation purposes which can be referred to in Section 5.3. Finally, Section 5.4 concludes the chapter.

## 5.1 Shrimp diet formulation study

This study aims to construct a diet formulation model for juvenile Whiteleg shrimp using Evolutionary Algorithm. Nutritional requirements for juvenile shrimp varies compared to adult shrimp especially for protein, which is the most expensive source in this animal diet. In order to obtain a minimum feed cost, sixteen single nutrients, two combined nutrients, and one ratio of nutrients, are constraints taken into consideration. Another constraint that also contributes to the shrimp diet is ingredient restrictions that must lie within the permitted range. A number of maximum ingredients in the ration is limited to fourteen; however users are able to choose the preferred ingredient to be included in the computation. The preferred total ingredient weight is the issue introduced in this study to help users such as nutritionists or farmers manage their expenditure well.

Then, after identifying the objective function and constraints of shrimp diet, a detailed procedure on the proposed hybrid Evolutionary Algorithm is discussed in the next section so that the procedure of a prototype that has been developed is better understood.

## 5.2 Development of diet formulation prototype through Evolutionary Algorithms

The diet formulation prototype is developed using hybrid Evolutionary Algorithm to fulfil the required constraints at a minimum cost. Some new procedures are introduced at the initialization phase, selection and crossover. In order to test each new operator, three models are developed with two sub-models for each model together with the existing establishes procedures. The simple pseudocode for an EA is shown in Figure 5.1.

i. Initialize solution in semi random procedure
   *Incorporate Power Heuristics*
ii. Evaluate each individual penalty value
iii. Select pair to mate using either *Roulette Wheel* or *Queen-Bee* or *Roulette – Tournament Selection operator*
iv. Do Crossover either using *One-Point* or *Average Crossover operator*
v. Do *Power Mutation*
vi. Repair operator using *Power Heuristics*
vii. Apply elitism operator
viii. Repeat step (iii) until step (vii) until a number of generations is reached

*Figure 5.1.* The Procedure for Hybrid EA

The diet formulation problem emerges when a combination of feed ingredients is to be decided in order to satisfy nutritional restriction at a minimum cost. From the industrial perspective, it is important to satisfy animal nutritional needs and at the same time increase the profit margin. The diet formulation prototype task is to find the best compromised solution from various combinations of ingredients in the database that gain the lowest possible penalty value.

The use of diet formulation prototype starts with the selection of ingredients by a user. The user is allowed to put and edit the ingredient data including price to meet the current fluctuations. The ability to edit the ingredient data would definitely ease the variability problem in the price of feed ingredients. This ability also allows the prototype to be used for other aquaculture or animals that require the same type of nutrients.

After the user is satisfied with all the inserted data, the prototype then generates a solution. When a termination criterion is met and the prototype stops its searching process, a list of ingredients with specific quantities along with the total price obtained appears at the prototype interface. The solution obtained satisfies all the hard constraints, which are maximum number of ingredients, total ingredient weight and crude protein. However, some soft constraints which are nutrient range and ingredient range might not satisfy. The procedures and implementation of the proposed enhanced EA activities are described in detail in the following sub-sections.

### 5.2.1 Representation structure

An appropriate structure of a solution representation is important to be decided to ensure the prototype could run smoothly. $X_i$ represents the number of ingredients predetermined by the user from a prototype interface. The number of alleles depends on the number of ingredients. In our proposed algorithm, fourteen ingredients are the maximum number of ingredients allowed to be computed at one time. Figure 5.2 shows an example of the encoded solution.

Ingredients, $X_i$

| $X_1$ | $X_2$ | $X_3$ | $X_4$ | $X_5$ | $X_6$ | $X_7$ | $X_8$ | $X_9$ | $X_{10}$ | $X_{11}$ | $X_{12}$ | $X_{13}$ | $X_{14}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 6 | 28 | 3 | 40 | 37 | 6 | 8 | 3 | 32 | 8.3 | 7 | 3 | 4.4 | 5 |

*Figure 5.2.* Representation Structure

The chromosome or individual represents one solution that consists of a combination of ingredient 1 until ingredient 14 with its specific weight chosen from a list of ninety-one ingredients. The representation is in the form of an array with the size of 1x14. The values of allele represent the weight of ingredients in the solution. For instance, $X_1$ equals to 6 means 6 kg of rice bran is placed as one of the solution's component.

### 5.2.2 Population initialization

The initial population is generated in a semi-random operation. The upper bound and the lower bound of the ingredients are stated as a limitation for the ingredients. The upper bound ($U_{X_i}$) and the lower bound ($L_{X_i}$) is the permitted quantity of each ingredient that is allowed to be included in the computation. The formula to generate an initial population is shown in the following figure. The number of chromosomes in the population remains the same until the stopping criteria are met. Figure 5.3 represents the initialization steps to generate a solution.

i. Generate an initial population by using formula
For $i$ =1, 2, 3,..., 14

$$IPOP_j = rand \ \lfloor L_{X_i}, U_{X_i} \rfloor \tag{5.1}$$

Where $L_{X_i}$ is ingredients' lower bound

$U_{X_i}$ is ingredients' upper bound

$j$ is the number of population

ii. Check feasibility of ingredients constraints for each chromosome
iii. If there is infeasibility in the ingredients constraints, repeat step (i) until feasible solution obtained for all ingredients.

*Figure 5.3.* Pseudocode for Generating Initial Population

#### 5.2.2.1 Penalty computation

After a solution is generated, a penalty value for each solution is computed. There are five conditions for computation of penalty value by checking the constraint violation, as follows:

i. at least one ingredient range constraint is violated.
ii. total ingredient weight is violated.
iii. at least one single nutrient constraint is violated.
iv. at least one nutrient combination is violated and
v. ratio constraint is violated.

If any of the above conditions is violated, the penalty value is set to be 1500 for a hard constraint. This big value is given to ensure all the violated solution will be removed from computation. This technique is done by Mohd Razali (2011) where 100 000 penalty is given for hard constraint. However, Mohd Razali (2011) only considered when one hard constraint is violated. Even if two or more hard constraints are violated, total penalty of 100 000 is also given. In this research, the penalty value is sum up of all constraints involved.

In this case, the constraints that are considered hard are total ingredient weight and crude protein in (iii) above. Total ingredient weight is essential to be fulfilling as it reflects farmers planning as ingredient is sold by its weight. However, a penalty for total ingredients weight is put in different range which can be referred in Figure 5.5. Crude protein is essential as protein is the most expensive nutrients sources compared to other nutrients. Crude protein is the estimated amount of the total protein content of a feed and is taken into consideration by almost all previous researchers. On the other hand, for soft constraints, the penalty values vary for every constraint based on expert opinions.

Pseudocodes in Figure 5.4 until Figure 5.8 represent five conditions to compute the penalty value. Subsequently, Figure 5.9 shows how the cost for each feed formulation is calculated.

i. At least one ingredients range constraint is violated.

All fourteen ingredients have their specific limitations with different values of lower bound and upper bound. The minimum (lower bound) and maximum (upper bound) requirement of the ingredients is given in percentage form as shown in Appendix B. If one or more of the ingredients obtained from the model solution is beyond these lower bound and upper bound, the solution violates the ingredient constraint. Table 5.1 shows the penalty values in relation to the violation of the ingredient range constraints.

Table 5.1

*List of Penalty Values for Ingredient Range Constraint*

| Ingredient | Penalty value |
|---|---|
| Rice bran, Malaysia, $X_1$ | 20 |
| Soybean meal, $X_2$ | 30 |
| Palm kernel cake, $X_3$ | 20 |
| Local fishmeal, $X_4$ | 30 |
| Wheat flour, $X_5$ | 20 |
| Wheat pollard, $X_6$ | 20 |
| Poultry meal, $X_7$ | 20 |
| Crude Palm Oil, $X_8$ | 20 |
| Imported fish meal, $X_9$ | 30 |
| Meat and bone meal, $X_{10}$ | 20 |
| Poultry by product, $X_{11}$ | 20 |
| Blood meal, $X_{12}$ | 20 |
| Krill meal, $X_{13}$ | 20 |
| Squid meal, $X_{14}$ | 20 |

A penalty value for each ingredient varies according to experts' opinion that is based on the digestibility of nutrients towards Whiteleg shrimp growth (The experts' background are explained in Section 4.1.2). The digestibility values are obtained from a council report by NRC (2011). For instance, if the values of a krill meal and local fish meal obtained from the model are greater than the allowable range, a penalty of 50 is given. The actual value of lower bound and upper bound is computed based on the total ingredient weight ($Y$) predefined by the user in the prototype interface. The calculation of lower bound and upper bound of each ingredient is as follows:

Lower bound:

$$L_{Xi} = \frac{\text{minimum restriction}}{100} \times Y \qquad (5.2)$$

Upper bound:

$$U_{Xi} = \frac{\text{maximum restriction}}{100} \times Y \qquad (5.3)$$

Where $Y$ is the total ingredient weight defined by the user

These actual values based on the total ingredient weights are shown in a separate column in the prototype interface once being calculated. The penalty value is then computed based on the actual lower bound and upper bound value. Figure 5.4 shows the pseudocode on how penalty value is computed for the ingredient constraint.

```
For Ingredient i = 1, 2, 3, …, 14, do
        If (X_i ≠ 0) and (X_i < L_{X_i} or X_i > U_{X_i})
            violation exist, set penalty based on Table 5.1;
        Else
            End
        End if
End for
```

*Figure 5.4.* Pseudocode for Ingredient Constraint Violation

The weight of all ingredients should lie within its permitted range. If the weight is exceed, which is either less than the lower bound value or more than the value of upper bound, then a specific penalty value is obtained. The ingredient's value also can be zero which means that the ingredient is not selected in the mix of solution.

ii. Total ingredient weight is violated.

The value of the total ingredient's weight or ration weight is determined by a user based on the requirement of the user which basically reflects the farm size. In order to obtain that particular weight, the sum of all selected ingredient weights should be computed. Figure 5.5 shows the procedure to calculate the total ingredient weight.

If $\left(\sum_{i=1}^{n} X_i\right) \geq Y - 0.5 \;\; or \left(\sum_{i=1}^{n} X_i\right) \leq Y + 0.5$ then

Set penalty = 0;

Else if

$\left(\sum_{i=1}^{n} X_i\right) \geq Y - 2 \;\; or \left(\sum_{i=1}^{n} X_i\right) \leq Y + 2$ then

Set penalty = 100;

Else if

$\left(\sum_{i=1}^{n} X_i\right) \geq Y - 5 \;\; or \left(\sum_{i=1}^{n} X_i\right) \leq Y + 5$ then

Set penalty = 200;

Else if

$\left(\sum_{i=1}^{n} X_i\right) \geq Y - 10 \;\; or \left(\sum_{i=1}^{n} X_i\right) \leq Y + 10$ then

Set penalty = 400;

Else

Set penalty = 1500;

End if

*Figure 5.5.* Pseudocode for Total Ingredient Weight Constraint Violation

In this procedure, *Y* is the weight determined by the user. Therefore, the total ingredient weight is supposed to equal to *Y*. However, in reality, the exact value equal to *Y* is difficult to obtain. Hence, we adjust some range of weight produced to increase the probability of gaining the targeted weight. These ranges are made based on five conditions as below:

a. within 0.5 kg from total ingredient weight

If the total ingredient weight lies in this range the solution is considered good where no penalty is given. For instance, let say *Y* is equal to 100 kg. If the total ingredient weight obtained by computation is 99.5 kg or 100.4 kg, the solution is acceptable.

b. within 2 kg from total ingredient weight

If the total ingredient weight is within this range, a penalty value of 100 is given for the solution. Sample solutions in this range are 98.4 kg and 101 kg, given *Y* is equal to 100 kg. A bigger penalty value is given as the total ingredient weight is important in the computation.

c. within 5 kg from total ingredient weight

This range accepts the value from 95 until 105, given *Y* is equal to 100 kg. The penalty value of 200 is given if the solution lies in this range.

d. within 10 kg from total ingredient weight

This is the biggest acceptable range where the acceptable value is from 90 to 110, given *Y* is equal to 100 kg. The penalty value of 400 is given if the solution lies in this range.

e. more than 10 kg from total ingredient weight

If the solution is too far from the actual value which is more than 10, the solution is not compromised at all and 1500 penalty value is given.

In this constraint, an acceptable difference of range value is until 10 kg. However, a big penalty value is set especially for the case of within 10 kg from the desired weight. The solution of more than 10 kg is rejected.

iii. At least one nutrient constraint is violated.

Conceptually, nutrient constraint is calculated by the same technique with ingredient constraint. The ingredient weight becomes the basis of getting the minimum and maximum value. A penalty value for crude protein is 1500, while for the approximate nutrients which are lipid, ash, fibre, calcium and phosphorus, penalty value is 40. A penalty value for amino acids is set at 30. In this case, if one or more nutrients are violated, a penalty value is given based on the nutrient's penalty value as shown in Table 5.2. For instance, if the solution from the model shows that ash and calcium violate their maximum value, then a penalty value of 80 is given for this solution.

Table 5.2

*List of Penalty Values for Nutrient Constraint*

| Nutrients | Penalty Value |
|---|---|
| Crude protein | 1500 |
| Lipid | 40 |
| Fiber | 40 |
| Ash | 40 |
| Calcium | 40 |
| Phosphorus | 40 |
| Arginine | 30 |
| Histidine | 30 |
| Isoleucine | 30 |
| Leucine | 30 |
| Lysine | 30 |
| Methionine | 30 |
| Phenylalanine | 30 |
| Threonine | 30 |
| Tryptophan | 30 |
| Valine | 30 |

Figure 5.6 shows the procedure for a single nutrient constraint violation. In this procedure, $N_{ki}$, $n_{ki}$, $L_{Nk}$ and $U_{Nk}$ is defined as below.

$N_{ki}$ = Total value of nutrient $k$ in a combination of ingredient $i$, $k$=1,2,…,16

$n_{ki}$ = nutrient $k$ consists in ingredient $i$

$L_{Nk}$ = lower bound of total value of nutrient $k$

$U_{Nk}$ = upper bound of total value of nutrient $k$

```
For (k = 1) do
        Calculate N_ki
        N_ki = (n_k1 . X_1) + (n_k2 . X_2) +…+ (n_ki . X_i)                    (5.4)
            If ( N_k < L_Nk or N_k > U_Nk)
                violation exist, set penalty = 1500;
            Else
                End
            End If
End for

For (k = 2, 3, 4, 5, 6)
        Calculate N_k
        N_ki = (n_k1 . X_1) + (n_k2 . X_2) +…+ (n_ki . X_i)
            If ( N_k < L_Nk or N_k > U_Nk)
                violation exist, set penalty = 40;
            Else
                End
            End If
End for

For (k = 7, 8, …, 16)
        Calculate N_k
        N_ki = (n_k1 . X_1) + (n_k2 . X_2) +…+ (n_ki . X_i)
            If ( N_k < L_Nk or N_k > U_Nk)
                violation exist, set penalty = 30;
            Else
                End
            End If
End for
```

*Figure 5.6.* Pseudocode for Single Nutrient Constraint Violation

iv. At least one nutrient combination is violated

A nutrient combination refers to the combination of two or more nutrients in specific quantities. In this study, two combinations of nutrients are considered to obtain a balanced diet, which are the combination of methionine ($n_{12}$) and cystine ($n_{18}$) and also the combination of phenylalanine ($n_{13}$) and tyrosine ($n_{17}$). A penalty value for both nutrient combinations is 20. Therefore, the total combination of both nutrients in a mix is obtained through the computation, as shown in Figure 5.7. In this computation, $n_{19}$, $N_{19}$, $n_{20}$ and $N_{20}$ is defined as below.

$n_{19}$ = combination of Methionine, $n_{12}$ and Cystine, $n_{18}$ in each ingredient,

$N_{19}$ = sum of Methionine, $n_{12}$ and Cystine, $n_{18}$ in all preferred ingredients

$n_{20}$ = combination of Phenylalanine, $n_{13}$ and Tyrosine, $n_{17}$ in each ingredient,

$N_{20}$ = sum of Phenylalanine, $n_{13}$ and Tyrosine, $n_{17}$ in all preferred ingredients

Calculate $N_{19}$

$$N_{19} = (n_{12}. X_1 + n_{18}. X_1) + (n_{12}. X_2 + n_{18}. X_2) + \ldots + (n_{12}. X_{14} + n_{18}. X_{14}) \quad (5.5)$$

If ($N_{19} < L_{(Met+Cys)}$ **or** $N_{19} > U_{(Met+Cys)}$)

violation exist, set penalty = 20;

Else

End

End if

Calculate $N_{20}$

$$N_{20} = (n_{13}. X_1 + n_{17}. X_1) + (n_{13}. X_2 + n_{17}. X_2) + \ldots + (n_{13}. X_{14} + n_{17}. X_{14}) \quad (5.6)$$

If ($N_{20} < L_{(Phe+Tyr)}$ **or** $N_{20} > U_{(Phe+Tyr)}$)

violation exist, set penalty = 20;

Else

End

End if

*Figure 5.7.* Pseudocode for Combination Nutrient Constraint Violation

In this procedure, the total nutrient value for both two nutrients is taken from each selected ingredient. For instance, to calculate $N_{19}$, the value of methionine and cystine in ingredient 1 until ingredient 14 is computed. The total value produced from the computation is then compared with the allowable range of the nutrient's combination. If the value lies within the lower bound ($L_{(Met+Cys)}$) and upper bound ($U_{(Met+Cys)}$), no penalties are given. Otherwise, a penalty value of 20 is given for the solution.

i. Ratio constraint is violated.

Based on data collection from experts and past studies, only one nutrient ratio is required for Whiteleg shrimp. Therefore, this study considers the ratio between calcium and phosphorus. For this constraint, if the ratio value is not equivalent to the lower limit of this nutrient ratio ($L_{ratio}$) and upper limit of this nutrient ratio ($U_{ratio}$), then 20 is put as a penalty. Otherwise, no penalty is given to the solution. The algorithm is shown in the following figure.

```
If (ratio calcium to phosphorus<L_ratio or ratio calcium to phosphorus>U_ratio) then
    violation exist, set penalty = 20;
Else
   End
End if
```

*Figure 5.8.* Pseudocode for Ratio Constraint Violation

The total penalty values for each solution are calculated based on the addition of the penalty values in all five conditions. Therefore, a total penalty value is a summation of the results from the computation in Figure 5.4 until Figure 5.8.

The computation of a total cost is based on a summation of the ingredients' weight by their price for one kg, as shown in Figure 5.9. In this procedure, $C_i$, $X_i$, $s$ and $n$ are defined as below.

$C_i$ is the cost of ingredient $i$ in one kilogram,

$X_i$ is the weight of the i*th* ingredient,

$s$ is the cumulative cost in a string of chromosome

and $n$ is the number of ingredient

For $i$ = 1,2,3,…..,14

Accumulate cumulative cost, *f(s)*

$$f(s) = \sum_{i=1}^{n} (X_i . C_i) \quad (5.7)$$

$$= (C_1 . X_1) + (C_2 . X_2) + \ldots + (C_n . X_n)$$

End for

*Figure 5.9.* Pseudocode for Cumulative Cost Calculation

### 5.2.2.2 Power Heuristics

A filtering operator called Power Heuristics is included as a repair operator in the methodology in which its function is to choose a better combination of feed ingredients based on certain performance, and reject the worst one. Subsequently, to cater for the strictness of nutrient and ingredient restriction, Power Heuristics is introduced at the initial part of the main method. This mechanism is useful when inappropriate ingredients need to be removed when an important constraint is violated, i.e., the total ingredient weight exceeds the required amount. The incorporation of this operator can at least reduce the initial penalty value, thus increasing the opportunity for acquiring a feasible solution.

This operator involves several similar steps adapted from the Power Mutation operator by Deep and Thakur (2007). The purpose of Power Mutation is to obtain a feasible solution within the constraint limitation. Thus, the formula of getting a new value of $X_{im}$ is designed by including a lower bound and upper bound value in the formula. The formula by Deep and Thakur (2007) is as below:

If ( $t < r$ ) : $$X_{im} = X_{ic} - s(X_{ic} - L_{Xi}) \tag{5.8}$$

Else if ( $t \geq r$ ): $$X_{im} = X_{ic} + s(U_{Xi} - X_{ic}) \tag{5.9}$$

Where $X_{im}$ = new ingredient *i* value

$X_{ic}$ = ingredient *i* value from crossover result

*s* = distribution value

$L_{X_i}$ = lower bound of ingedient *i*

$U_{X_i}$ = upper bound of ingredient *i*

In our case, Power Heuristics aims to reject some ingredients and then repair the solution to obtain a lower penalty value. Therefore, we introduced the formula in (iv) from Figure 5.10 so that a new value of ingredients from this Power Heuristics operator ($X_{if}$) can be either zero or slightly different from the original value. This formula differs from Deep and Thakur (2007) in the sense that the weight of the selected ingredient *i* is in the range of $\pm 1$ of its value or similarly $(X_i - 1 < X_i < X_i + 1)$. This step is motivated by experts' opinions that inquired about the number of ingredients considered in this research is too many. Furthermore, we realized that other study also used less than fourteen ingredients for shrimp feed such as in Htun et al. (2005). Figure 5.10 shows the procedure involved in Power Heuristics, while Figure 5.11 shows Power Mutation by Deep and Thakur (2007).

Both pseudocode is put together so that we can see the different more clearly. Then, an example of real Power Heuristics in shrimp diet formulation is shows in Figure 5.12.

i. Generate uniform random number, *r* between [0,1]

ii. Get *t* using formula:

$$t = \frac{X_i - L_{Xi}}{U_{Xi} - X_i} \tag{5.10}$$

where $X_i$ : ingredient $i$ value from initial solution
$L_{Xi}$ : lower bound of ingredient $i$
$U_{Xi}$ : upperbound of ingredient $i$

iii. Compare value *t* with *r* and determine which one is greater

iv. Find new value of $X_{if}$ by comparing *r* with *t*

If $r > t$ :

Then new value of $X_{if} = 0$

If $t \geq r$ :

Then generate new value of $X_{if}$ by the formula: $\left(X_i - 1 < X_i < X_i + 1\right)$ (5.11)

v. Repeat step (ii) to (iv) for other allele and also for other infeasible individuals

vi. Calculate new penalty value for the solution

*Figure 5.10.* Pseudocode for Power Heuristics

i. Generate uniform random number $s_1$ and *r* between [0, 1]

ii. Get *s* using formula:

$$s = p\,(s_1)^{p-1}$$

where $s_1$ is random number
$p$ is index of Power distribution

iii. Get *t* using formula:

$$t = \frac{X_{ic} - L_{Xi}}{U_{Xi} - X_{ic}}$$

where $X_{ic}$ : ingredient $i$ value from crossover stage
$L_{Xi}$ : lower bound of ingredient $i$
$U_{Xi}$ : upperbound of ingredient $i$

If ( $t < r$ )

Then generate new value of $X_{im}$ by the formula

$$X_{im} = X_{ic} - s(X_{ic} - L_{Xi}) \tag{5.8}$$

Else if ( $t \geq r$ )

Then find new value of $X_{im}$ by the formula

$$X_{im} = X_{ic} + s(U_{Xi} - X_{ic}) \tag{5.9}$$

iv. Repeat step (iii) to (iv) until all alleles in chromosome *i* are mutated

v. Calculate new penalty value for the solution.

*Figure 5.11.* Pseudocode for Power Mutation

Example:

Before:

| $X_1$ | $X_2$ | $X_3$ | $X_4$ | $X_5$ | $X_6$ | $X_7$ | $X_8$ | $X_9$ | $X_{10}$ | $X_{11}$ | $X_{12}$ | $X_{13}$ | $X_{14}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 6 | 28 | 3 | 40 | 37 | 6 | 8 | 3 | 50 | 11 | 13 | 3.4 | 4 | 5 |

After:

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 27.3 | 4 | 0 | 35.9 | 6.32 | 0 | 4 | 0 | 10.3 | 0 | 4 | 3 | 4.9 |

*Figure 5.12.* An Example of Power Heuristics Output

Figure 5.12 shows the output from the Power Heuristics operator. A new solution after performing the operator shows that the total ingredients are reduced to nine when five ingredients are removed. These ingredients are rice bran ($X_1$), local fish meal ($X_4$), poultry meal ($X_7$), imported fish meal ($X_9$), and poultry by product ($X_{11}$). By reducing some ingredients, the total weight for this solution is also reduced from 217.4 kg to 99.72 kg. The new solution fulfils the ration weight constraint, and thus has a less penalty value.

### 5.2.3 Parents selection operator

In this research, three parent selection operators are experimented. These operators are Roulette Wheel Selection (RWS), Queen-Bee Selection and Roulette-Tournament Selection. The first two are established operators, while the third one is a newly proposed operator based on the combination of RWS and Binary Tournament Selection. The detailed mechanism of each selection technique is discussed in the next subsections.

#### 5.2.3.1 Roulette Wheel Selection

In reference to the objective function, a shrimp feed formulation is a minimization problem. Thus, we create a new penalty value to suit the Roulette Wheel Selection (RWS) mechanism. The proportion is changed to ensure that the value is fair. In other words, the lower the minimum value, the greater the penalty value, which means more chances to be elected as a parent. The computation to obtain a new penalty value is called Roulette Penalty Value (RPV), which is given below. This computation is originally proposed by Mohd Razali (2011). But we later simplify the step. The detailed computation is presented in Figure 5.13.

$$RPV = (Maximum\ penalty - Minimum\ penalty) + Current\ penalty \qquad (5.12)$$

i. Determine the maximum penalty (maxpenalty) in the population
ii. Determine the minimum penalty (minpenalty) in the population.
iii. Get the different between maximum penalty and minimum penalty (penaltydiff)

Penaltydiff = maxpenalty – minpenalty

iv. Get the Roulette Fitness Value for each individual in the population

RPV = Penaltydiff + Current individual penalty

*Figure 5.13.* Pseudocode for Roulette Penalty Calculation

The RWS operator begins by selecting parents from a population. The number of parents depends on the crossover probability rate ($P_c$). In shrimp diet formulation, the probability is set at 0.6 based on the experimentation done in Section 6.1. The detailed RWS procedure is shown in Figure 5.14.

i. Sum up the total RFV, *sum*RPV

ii. Calculate *r*RFV by dividing RPV with *sum*RPV

$$rRPV = RPV/sumRPV \qquad (5.13)$$

iii. Calculate *c*RPV(*i*+1) by adding the current *c*RPV (*i*) with the previous *r*RPV (*i*+1)

$$cRPV\ (i+1) = cRPV(i) + rRPV\ (i+1) \qquad (5.14)$$

iv. Generate random number (*Ri*) for each individual *i*.

v. Compare the *Ri* value with the first value of RPV.

If (*Ri* ≤ the value of the first RPV)

Select individual (*i*) to be put in a new population.

Else if (*Ri*> the value of the first RPV)

Compare *Ri* value with individual *i*+1. Select individual *i*+1 if *Ri* value is between these range

*Figure 5.14.* Pseudocode for Roulette Wheel Selection

#### 5.2.3.2 Queen-Bee Selection

In a Queen-Bee (QB) Selection operator, the concept of queen bee is adopted only when the fittest individual is chosen as a parent. The drone or the partner for the queen bee is chosen randomly from any individuals in the population. The following figure shows the steps involved in this selection operator.

i. Arrange all individuals in descending order
ii. Take the best individual at the top of the list as parent 1
iii. Choose 1 individual randomly from the rest of the individual in the population to be parent 2

*Figure 5.15.* Pseudocode for Quee-Bee Selection

#### 5.2.3.3 Roulette-Tournament Selection

In this operator, a Roulette Wheel Selection is first conducted to select a number of chromosomes based on a percentage predetermined at the beginning of the process. From the experimentation that has been conducted, 60% is used as a percentage. Therefore, 18 chromosomes are selected. Chromosomes are then chosen by a Binary Tournament Selection mechanism.

i. Do the same procedure of RWS as in Figure 5.13 to the whole population.
ii. Choose two individuals from RWS randomly.
iii. Compare the penalty of the two individuals
iv. Choose the better individual with lower penalty value as parent 1
v. Repeat step (ii) until step (iv) to choose parent 2

*Figure 5.16.* Pseudocode for Roulette-Tournament Selection

### 5.2.4 Crossover operator

In this research, the proposed EA algorithm is experimented on two types of operator. The first is a One-Point Crossover and the second one is a modification of One-Point Crossover, in which Average Crossover is introduced.

#### 5.2.4.1 One-Point Crossover

A Single-Point or One-Point Crossover is an established concept introduced by Holland in 1975 (Booker et al., 2000), which means only one point is chosen as a cutting point. After that, allele in both individuals is interchanged at the chosen point. In this research, the middle point is selected as the cutting point. Figure 5.17 represents the steps involved in One-Point Crossover while Figure 5.18 illustrates the real One-Point Crossover in a shrimp diet formulation problem.

i. Choose midpoint as cutting point in parent 1 and parent 2
   Midpoint = chromosome length/2 = 14/2 = 7 (5.15)
ii. Copy left segment of parent 1 to child 1 and right segment of parent 2 to child 1.
iii. Copy left segment of parent 2 to child 2 and right segment of parent 1 to child 2.
iv. Calculate total penalty value of $C_1$ and $C_2$ as in Figure 5.4 until Figure 5.8.
v. Update current penalty value

*Figure 5.17.* Pseudocode for One-Point Crossover

Example: One-Point Crossover to produce new solution

Parent 1:

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 6 | 28 | 3 | 40 | 37 | 6 | 8 | 3 | 50 | 11 | 13 | 4 | 4 | 5 |

Parent 2:

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 5 | 16 | 4 | 7 | 30 | 14 | 14 | 3 | 17 | 8 | 9 | 3 | 3 | 4 |

Child 1:

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 6 | 28 | 3 | 40 | 37 | 6 | 8 | 3 | 17 | 8 | 9 | 3 | 3 | 4 |

Child 2:

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 5 | 16 | 4 | 7 | 30 | 14 | 14 | 3 | 50 | 11 | 13 | 4 | 4 | 5 |

*Figure 5.18.* An Example of One-Point Crossover

### 5.2.4.2 Average Crossover

Average Crossover is a new concept that we introduce based on the midpoint of the ingredient values between two parents. The aim of this operator is to fully utilize the fitter parent ingredient's value. As mention earlier, the shrimp ingredients are represented by real-valued alleles in a chromosome of the EA. Therefore, an alternative crossover operator similar to the One-Point crossover is deemed suitable for the performance of the whole EA process.

A midpoint value from two parents obtained as a result of the selection operator is calculated as child 1 and child 2. One cutting point is selected in the middle of a chromosome as in equation 5.13. Subsequently, both parents are crossed over at the selected point. Figure 5.19 shows the pseudocode for obtaining the offspring using

Average Crossover. The illustration of this mechanism in shrimp feed formulation can be referred to in Figure 5.20.

i. Calculate the midpoint of all allele in parent 1 and parent 2

$$(X_{i1} + X_{i2})/2 \tag{5.16}$$

Where $X_{i1}$ = ingredient value of allele 1

$X_{i2}$ = ingredient value of allele 2

ii. Repeat step (i) for all alleles

iii. Choose midpoint as cutting point in parent 1 and parent 2

Midpoint = chromosome length/2 = 14/2 = 7

iv. Put the value of left segment of midpoint to left segment of child 1 and right segment of midpoint to right segment of child 2

v. Copy right segment of better parent to child 1 and left segment of better parent to child 2

vi. Calculate total penalty value of $C_1$ and $C_2$ as in Figure 5.4 until Figure 5.8.

vi. Update current penalty value

*Figure 5.19.* Pseudocode for Average Crossover

Example: Average Crossover to produce new solution

Parent 1:

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 6 | 28 | 3 | 40 | 37 | 6 | 8 | 3 | 50 | 11 | 13 | 4 | 4 | 5 |

Parent 2:

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 5 | 16 | 4 | 7 | 30 | 14 | 14 | 3 | 17 | 8 | 9 | 3 | 3 | 4 |

Child 1:

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 5.5 | 22 | 3.5 | 23.5 | 33.5 | 10 | 11 | 3 | 50 | 11 | 13 | 4 | 4 | 5 |

Child 2:

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 6 | 28 | 3 | 40 | 37 | 6 | 8 | 3 | 33.5 | 9.5 | 11 | 3.5 | 3.5 | 4.5 |

*Figure 5.20.* An Example of Average Crossover

### 5.2.5 Mutation operator

In this study, Power Mutation is chosen as mutation operator that has an advantage to search for more search space. After that, Power Heuristics takes place before the end of each generation.

#### 5.2.5.1 Power Mutation

Mutation operator is one of the most important operators in an EA. In shrimp feed modelling, we adopt Power Mutation operator introduced by Deb and Thakur (2007) based on the concept of power distribution ($p$). A $p$ value of 0.25 is employed in this research as proposed by Deb and Thakur (2007). This operator makes use of the value obtained from a crossover operator in order to find a new value of $X_{im}$. Figure 5.21 shows detailed steps involved in the Power Mutation procedure. Figure 5.22 shows an example of this operator in a shrimp feed formulation problem.

i. Generate uniform random number $s_1$ and $r$ between [0, 1]

ii. Get $s$ using formula:

$s = p\,(s_1)^{p-1}$

where $s_1$ is random number

$p$ is index of Power distribution

iii. Get $t$ using formula:

$$t = \frac{X_{ic} - L_{Xi}}{U_{Xi} - X_{ic}}$$

where
$X_{ic}$ : ingredient $i$ value from crossover stage
$L_{Xi}$ : lower bound of ingredient $i$
$U_{Xi}$ : upperbound of ingredient $i$

If ( $t < r$ )

Then generate new value of $X_{im}$ by the formula

$$X_{im} = X_{ic} - s(X_{ic} - L_{Xi}) \tag{5.8}$$

Else if ( $t \geq r$ )

Then find new value of $X_{im}$ by the formula

$$X_{im} = X_{ic} + s(U_{Xi} - X_{ic}) \tag{5.9}$$

iv. Repeat step (iii) to (iv) until all alleles in chromosome $i$ are mutated.

v. Calculate new penalty value for the solution.

*Figure 5.21.* Pseudocode for Power Mutation

Example: Power Mutation to produce new solution

Before:

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 7.5 | 17.5 | 4 | 27.5 | 35 | 10 | 10 | 3.5 | 37.5 | 10 | 10 | 4 | 4 | 4 |

After:

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 10.40 | 23.78 | 3.41 | 14.43 | 32.09 | 7.09 | 7.09 | 2.63 | 24.43 | 7.10 | 7.10 | 3.42 | 3.42 | 3.42 |

*Figure 5.22.* An Example of Power Mutation

#### 5.2.5.2 Power Heuristics

At this point, after performing Power Mutation, a Power Heuristics procedure is again activated to filter the unwanted combinations of ingredients by adjusting the value of ingredients in order to meet the constraints requirement especially for hard constraint. The pseudocode for Power Heuristics is as shown in Figure 5.23 is performed.

i. Generate uniform random number, *r* between [0,1]

ii. Get *t* using formula:

$$t = \frac{X_{im} - L_{Xi}}{U_{Xi} - X_{im}} \tag{5.17}$$

where $X_{mi}$ : ingredient $i$ value from mutation stage
$L_{Xi}$ : lower bound of ingredient $i$
$U_{Xi}$ : upperbound of ingredient $i$

iii. Compare value *t* with *r* and determine which one is greater

iv. Find new value of $X_{if}$ by comparing *r* with *t*

If $r > t$ :

Then new value of $X_{if} = 0$

If $t \geq r$ :

Then generate new value of $X_{if}$ by the formula: $\left(X_{im} - 1 < X_{im} < X_{im} + 1\right)$ (5.18)

v. Repeat step (ii) to (iv) for other allele and also for other infeasible individuals

vi. Calculate new penalty value for the solution

*Figure 5.23.* Pseudocode for Power Heuristics

### 5.2.6 Steady-State Selection procedure

After one iteration of the complete sequenced procedures is completed, the process is repeated until a number of generations is reached ($G_n$). In order to start a new generation, an elitism mechanism is applied to select the best individual from the current generation and bring forward to the next generation. The pseudocode for this mechanism can be referred to in Figure 5.24.

i. Find the number of new individuals to be completed in the generation.

Number of new individuals, *n* = original population size – number of current individuals that have been generated based on crossover probability

ii. Sort other individuals in current generation in ascending order of its fitness

iii. Get *n* number of individuals with *n* smallest fitness values from the sorted individuals

iv. The set of individuals with the same population size is reshuffled for randomness and thus ready for the next generation (repeat steps in Section 5.2.3 to 5.2.5)

*Figure 5.24.* Pseudocode for Steady-State Selection

### 5.2.7 Termination Procedure

The process of searching continues in continuous generations until a predetermined generation number is reached. When the process stops, the result consisting of a list of ingredients in specific quantities, total ingredient's weight and total cost is shown in the prototype's interface. For ease of understanding the results, the shrimp diet formulation prototype is discussed in this section on the model implementation.

## 5.3 Model implementation

A prototype is developed with some buttons to ease the process of editing the ingredients and nutrient constraints. An 'Add Restriction' function is to add new ingredients, while a 'Remove Restriction' function is to delete the existing ingredient(s) out from the computation. However, the prototype only allows until fourteen ingredients. A user may choose ingredients based on the list of ingredients that can be viewed in the 'Ingredients Table' interface. A nutrient constraint can be edited on the 'Nutritional Requirements' interface. 'Nutritional Combine Requirements' and 'Nutrient Ratios' are functions to edit the minimum and maximum restriction of nutrient combination constraints and nutrient ratio constraint, respectively.

The proposed hybrid EA and its three variants with three sub-models respectively are programmed using Java language on a personal computer of which the processor is Pentium ® 4 CPU 3.20 GHz 3.19 GHz and the memory size is 2.99 GB with Windows XP Professional operating system.

Initially, the user has to choose the recommended ingredients from a database that consists of ninety-one ingredients. Next, the recommended value of the total ingredient weight should be determined by the user so that the prototype can produce the weights based on the user's request (see Figure 5.25). In order to start the feed formulation process, the user must click on the button 'Formulate' to run the algorithm. In several minutes, the process stops and the user can view the result at the 'Formula Processing' interface, as shown in Figure 5.25 below. This figure shows the insertion interface where the user needs to put the preferred ration weight before the algorithm can run.

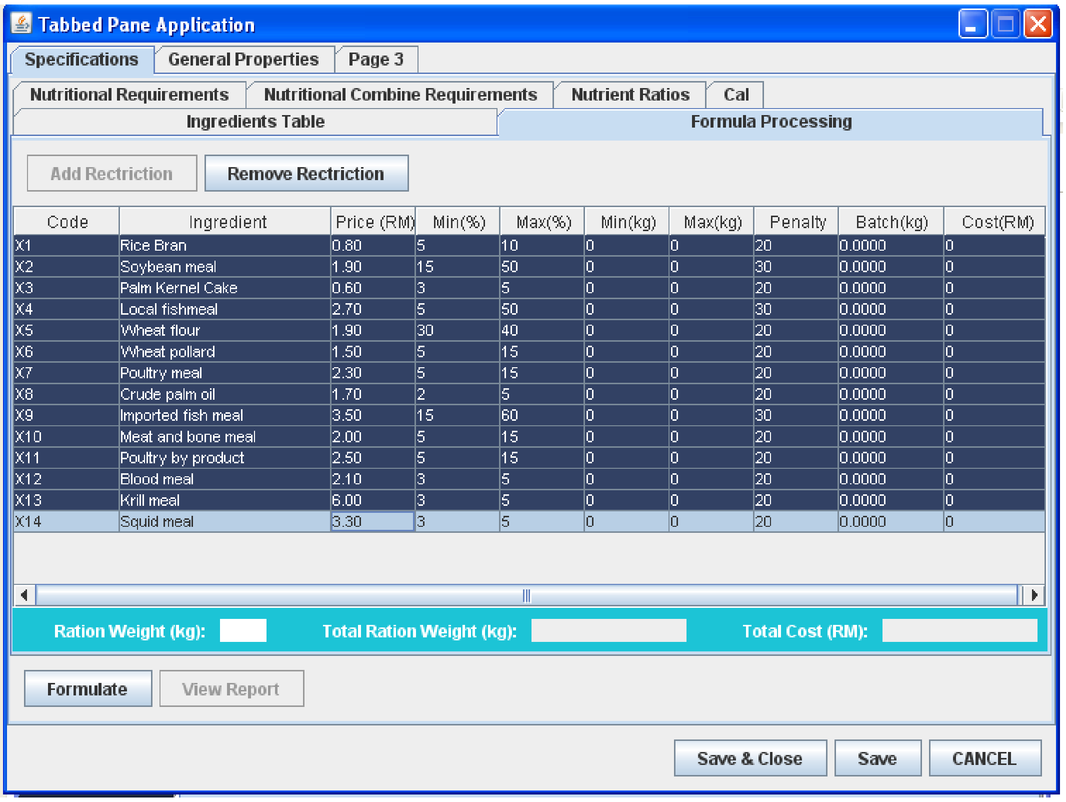


*Figure 5.25.* Ration Weight Insertion Interface

### 5.4 Summary

This chapter has described each operator involved in the proposed algorithm and its variants to achieve the two specific objectives. The second specific objective is to construct a new filtering heuristics known as Power Heuristics as part of the initialization operator that is capable of filtering some combinations of ingredients from a selected database of choices, which could lead to a potentially poor solution. Meanwhile, the third specific objective is to construct a new crossover operator known as Average Crossover that is able to produce a potentially good solution. In this chapter also a model implementation is illustrated on how to use the diet formulation prototype. The experimentation and results obtained from this study are discussed in the next chapter.

# CHAPTER 6

# RESULTS AND DISCUSSIONS

The testing, validation and experimentations of the proposed algorithm and its variants are discussed in this chapter. These activities were carried out to evaluate the performance of the algorithms. The experimentation includes testing on the performance of the algorithms using different selection and crossover operators. All experimentations were carried out using real data for juvenile Whiteleg shrimp as explain in the following section.

## 6.1 Identification of basic data

A set of fourteen ingredients ($X_1$ to $X_{14}$) was used in this diet formulation as shown in Table 6.1. The total ingredient weight was fixed at 100 kg. The total ingredient weight is the total weight of all selected ingredients to be mixed in the shrimp diet. In this research, the total ingredient weight was based on user preferences where the prototype was obtained at the requested weight by the user. The penalty values were given based on certain ranges of the total ingredient weight obtained from the prototype. The difference until 10 kg of weight is accepted with a maximum penalty of 400. If the difference is more than 10 kg, the solution is rejected.

Table 6.1

*Information on Ingredients*

| **Ingredient** | **Minimum (kg)** | **Maximum (kg)** | **Price (RM/kg)** | **Penalty value** |
|---|---|---|---|---|
| Rice bran, Malaysia, $X_1$ | 5 | 10 | 0.80 | 20 |
| Soybean meal, $X_2$ | 15 | 50 | 1.90 | 30 |
| Palm kernel cake, $X_3$ | 3 | 5 | 0.60 | 20 |
| Local fishmeal, $X_4$ | 5 | 50 | 2.70 | 30 |
| Wheat flour, $X_5$ | 30 | 40 | 1.90 | 20 |
| Wheat pollard, $X_6$ | 5 | 15 | 1.50 | 20 |
| Poultry meal, $X_7$ | 5 | 15 | 2.30 | 20 |
| Crude Palm Oil, $X_8$ | 2 | 5 | 1.70 | 20 |
| Imported fish meal, $X_9$ | 15 | 60 | 3.50 | 30 |
| Meat and bone meal, $X_{10}$ | 5 | 15 | 2.00 | 20 |
| Poultry by product, $X_{11}$ | 5 | 15 | 2.50 | 20 |
| Blood meal, $X_{12}$ | 3 | 5 | 2.10 | 20 |
| Krill meal, $X_{13}$ | 3 | 5 | 6.00 | 20 |
| Squid meal, $X_{14}$ | 3 | 5 | 3.30 | 20 |

Some operators such as Roulette Wheel Selection (RWS) and Power Mutation should be run to find the best probability value for each operator. The number of generation needs to be tested to obtain the most appropriate value that gives the least average penalty value in its run. It is important to average the results of several runs as EA has many random components. Table 6.2 depicts the parameters used in our models as obtained from various experimentations.

Table 6.2

*Parameters Value*

| **Properties** | **Value** |
|---|---|
| Crossover probability | 0.60 |
| Power Mutation index | 0.25 |
| Generation number | 200 |
| Number of runs | 30 |

The index of Power Mutation, $p$ is 0.25, that is based on a range of 0 to 1, as recommended by Deep and Thakur (2007), whereas crossover with probability of 0.6

was chosen based on the best-so-far penalty or solution obtained from various experimentations (refer to Table 6.3). Average run time was increased accordingly with the increment of crossover probability value, but the number of infeasible solution for all crossover probabilities gave approximately the same results. The average penalty value for the crossover with probability of 0.1 gave the lowest value, i.e., 520. However, the best-so-far penalty of 300 was obtained from both crossover with probability of 0.6 and 0.8. In terms of the number of infeasible solution, no solution was infeasible for crossover with probability of 0.6, while three infeasible solutions was obtained from crossover with probability of 0.8.

Table 6.3

*Results of EA Performance with Suggested Crossover Probabilities*

| **Crossover probability** | **Best-so-far penalty** | **Average penalty** | **Average run time (sec.)** | **Number of infeasible solution** |
|---|---|---|---|---|
| 0.1 | 380 | 520.0000 | 3991.7500 | 5/30 |
| 0.2 | 570 | 593.0000 | 5484.5000 | 5/30 |
| 0.3 | 530 | 560.0000 | 7140.0045 | 4/30 |
| 0.4 | 430 | 585.0000 | 9585.6925 | 1/30 |
| 0.5 | 310 | 530.3333 | 9704.3478 | 4/30 |
| 0.6 | 300 | 547.0000 | 12091.5000 | 0/30 |
| 0.7 | 340 | 556.6667 | 12472.7273 | 0/30 |
| 0.8 | 300 | 595.0000 | 12840.0000 | 3/30 |
| 0.9 | 540 | 558.5714 | 14625.0342 | 5/30 |
| 1.0 | 410 | 592.0000 | 16328.5714 | 3/30 |

As a result, with the established selection type, that is the Roulette Wheel Selection and crossover type of One-Point, the EA-RWS-One-Pt Model was run in different sizes of population to find the most suitable population size to be considered. The population size must be large enough so that it can support sufficient genetic variation (Chinneck, 2006). Besides, if the population size is too small, not enough points in the search space are sampled, which could lead to possible premature convergence. However, the size of population cannot be too large as the computation time will become too long (Chinneck, 2006).

From the experimentation result in Table 6.4, population sizes of 60, 70 and 80 gave equally the same best-so-far penalty value of 300. However, the population size of 60 was the most appropriate to be used in our problem due to no infeasible solution was obtained and a relatively shorter run time was observed compared to that of the others. The time is about 201.525 minutes. As a result, for further experimentations in other subsequent subsections, we used 60 in the population size.

Table 6.4

*Results at Different Sizes of Population*

| **Population size** | **EA-RWS-One-Pt Model** | | | |
|---|---|---|---|---|
| | **Best-so-far penalty** | **Average penalty** | **Average run time (sec.)** | **Number of infeasible solution** |
| 10 | 400 | 605.2381 | 1710.0430 | 5/30 |
| 20 | 450 | 579.0909 | 3402.1820 | 6/30 |
| 30 | 340 | 554.5833 | 5144.0216 | 4/30 |
| 40 | 340 | 598.0000 | 7281.1232 | 4/30 |
| 50 | 380 | 466.2500 | 9572.0545 | 3/30 |
| 60 | 300 | 547.0000 | 12091.5000 | 0/30 |
| 70 | 300 | 538.7500 | 13379.2208 | 1/30 |
| 80 | 300 | 546.2312 | 14943.2607 | 0/30 |

### 6.2 Experimentation with initialization operator

The semi-random initialization operation was implemented to find potential solutions in the initial stage. Two initialization procedures were experimented and thus suggesting the best initialization operator to be used in our proposed model. The first procedure employed a formula using lower bound and upper bound values of weight of each ingredient, as described in Section 5.2.2 (refer to formula 5.1). The second procedure also employed the same formula as in formula 5.1, but with the inclusion of Power Heuristics, as indicated in Section 5.2.2.2 for filtering purposes. The experimentations among these two initialization procedures were made and the results are shown in Table 6.5. In these experimentations, other operators were controlled with selection using RWS, crossover using the One-Point Crossover and mutation using Power Mutation. The parameter values used in this experimentation are the same as in Table 6.2.

Table 6.5

*Results for Different Initialization Procedure*

| **Model** | **Best-so-far penalty** | **Average penalty** | **Average run time (sec.)** | **Number of infeasible solution** |
|---|---|---|---|---|
| EA-SR | NA | All solution infeasible | 9942.8571 | 30/30 |
| EA-PH | 300 | 547.0000 | 12091.5000 | 0/30 |

The EA-SR is an evolutionary model with the semi-random initialization procedure, while EA-PH is another evolutionary model with the inclusion of Power Heuristics operator. EA-SR gave an infeasible solution in every run, whereas EA-PH Model showed that no solutions were infeasible as can be seen in Table 6.5. This experiment shows that Power Heuristics allows unnecessary ingredients to be filtered out of the system. As a result, the total selected ingredients in the system might be less than fourteen. A sample solution is shown in Table 6.6. The table shows that only eight ingredients were selected in the solution with the total weight of 100 kg. The total price for this feed combination was RM 217.08. Table 6.7 then shows the nutrients value for each solution obtained from EA-PH Model. The result from this experiment concludes that for every EA model experimented in this research, Power Heuristics must be included in the initialization procedure due to its better performance.

Table 6.6

*A Sample Solution of EA-PH Model*

| **Ingredient** | **Minimum (kg)** | **Maximum (kg)** | **Quantity (kg)** |
|---|---|---|---|
| Soybean meal, $X_2$ | 15 | 50 | 42.5860 |
| Palm kernel cake, $X_3$ | 3 | 5 | 4.0431 |
| Wheat pollard, $X_6$ | 5 | 15 | 14.3030 |
| Poultry meal, $X_7$ | 5 | 15 | 13.1132 |
| Crude Palm Oil, $X_8$ | 2 | 5 | 4.0231 |
| Poultry by product, $X_{11}$ | 5 | 15 | 12.1350 |
| Krill meal, $X_{13}$ | 3 | 5 | 4.6120 |
| Squid meal, $X_{14}$ | 3 | 5 | 5.2360 |
| *Total weight (kg)* | | | 100.0514 |
| *Total cost (RM)* | | | 217.08 |

Table 6.7

*Nutrients Value for EA-PH Model Solution*

| **Nutrients** | **Minimum** | **Maximum** | **Quantity** |
|---|---|---|---|
| Crude Protein, % | 38.00 | 45.00 | 44.1391 |
| Lipid, % | 0.08 | 0.18 | 8.5411 |
| Fibre, % | 0 | 4.00 | 4.6652 |
| Ash, % | 0 | 15.0 | 8.0950 |
| Calcium, % | 0 | 2.30 | 0.6631 |
| Phosphorus, % | 0.30 | 0.70 | 0.5039 |
| Arginine, % | 2.2 | 2.32 | 2.8308 |
| Histidine, % | 0.6 | 0.84 | 0.9641 |
| Isoleucine, % | 1.0 | 1.33 | 6.0316 |
| Leucine, % | 1.7 | 2.16 | 7.5201 |
| Lysine, % | 1.55 | 1.65 | 2.6221 |
| Methionine, % | 0.7 | 0.96 | 0.7256 |
| Phenylalanine, % | 1.4 | 1.6 | 1.9540 |
| Threonine, % | 1.3 | 1.44 | 0.4332 |
| Tryptophan, % | 0.2 | 0.32 | 1.6085 |
| Valine, % | 1.4 | 1.6 | 2.1226 |
| Methionine + Cystine, % | 1.0 | 1.44 | 1.3287 |
| Phenylalanine + Tyrosine, % | 2.7 | 7.1 | 2.8917 |
| Calcium:Phosphorus, % | 0.7692 | 0.7692 | 1.3159 |

## 6.3 Experimentation with different selection operators

Subsequent experimentations of EA with three selection operators, i.e., Roulette Wheel Selection (RWS), Queen-Bee (QB) Selection, and Roulette-Tournament (RT)

Selection were conducted. These selection operators were tested, while other EA operators remain the same. These controlled operators were initialization with incorporation of Power Heuristics, One-Point Crossover and Power Mutation. The parameter values used in this experimentation is the same as in Table 6.2. In this section, the process flow of the EA Models is discussed in Subsection 6.3.1. Then, the overall comparison on the performance of these selection operators can be referred to in Subsection 6.3.2.

### 6.3.1 The process flows of different selection operators

The generic process flow is shown in the Figure 6.1.

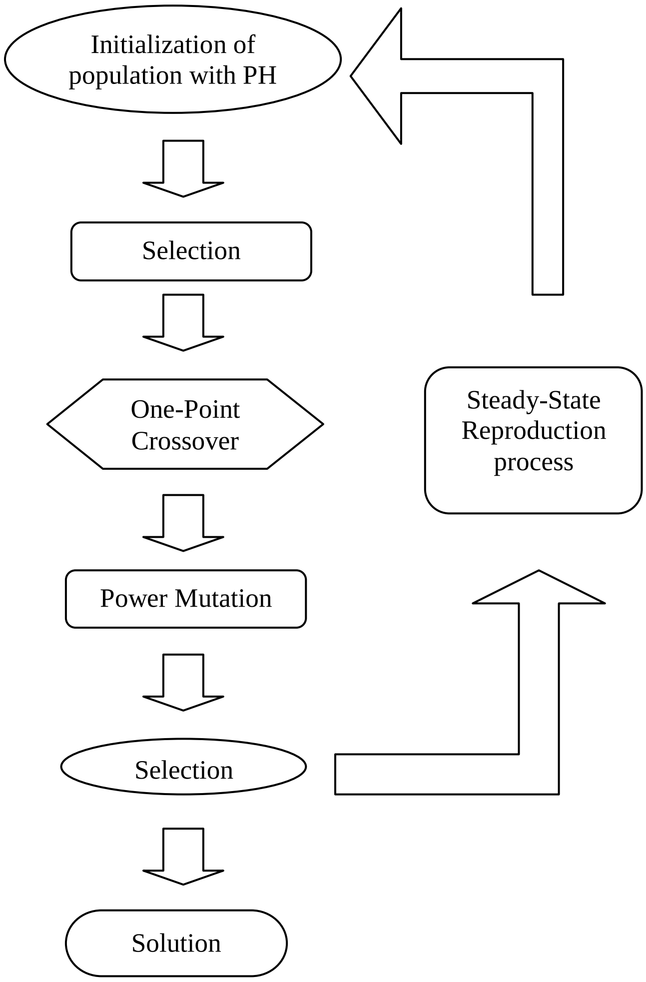



*Figure 6.1.* The Process Flow of EA Model

At first, an EA-PH-RWS-One-Pt Model was tested and reported in this subsection. The process flow is as shown in Figure 6.1 with the controlled operators of standard One-Point Crossover and Power Mutation. Then, the experimentation was altered when the RWS operator was replaced by the Queen-Bee (QB) Selection operator. This second altered model is known as the EA-PH-QB-One-Pt Model. The QB Selection concept is mainly about finding the least penalty value in each generation to obtain the best-so-far solution faster based on a penalty value obtained. The experimentation was again altered when the RWS operator was replaced by the Roulette-Tournament (RT) Selection operator. This model is known as the EA-PH-RT-One-Pt Model. The concept of Roulette-Tournament is about combining the advantages of both RWS and Tournament Selection to obtain efficient selection performance towards achieving the best solution. Figure 6.2 exhibits sample solutions obtained from all three models.

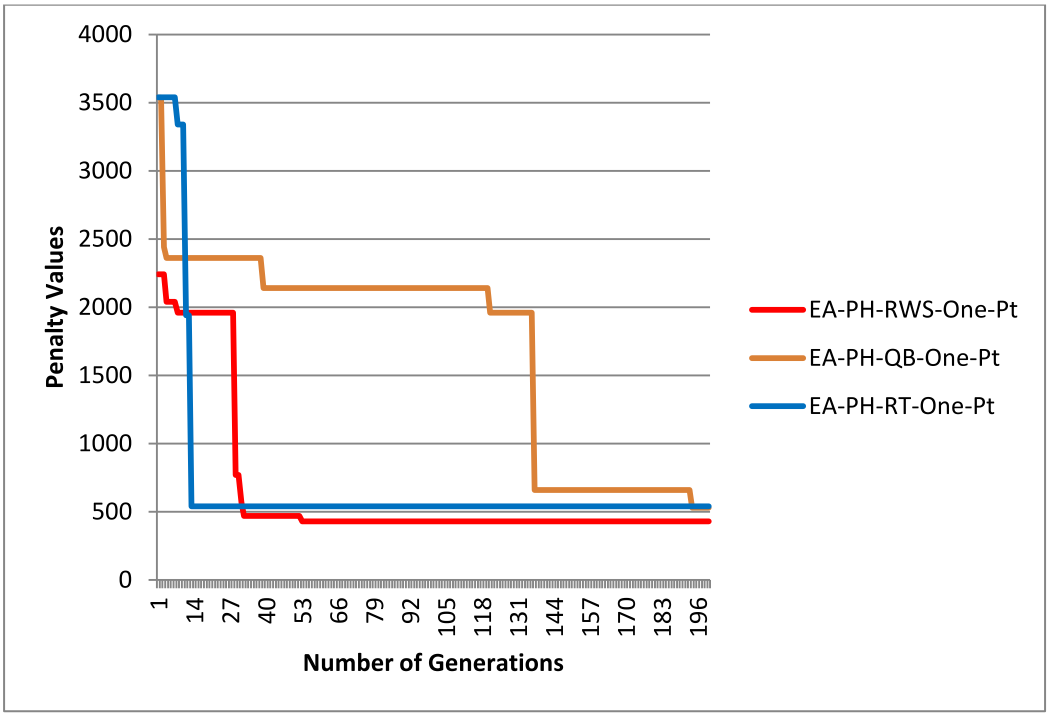



*Figure 6.2.* Sample Solutions of all Three Models with Different Selection Operators

Figure 6.2 shows sample solutions obtained from all the three EA models which are EA-PH-RWS-One-Pt, EA-PH-QB-One-Pt and EA-PH-RT-One-Pt. The red colour graph represents the EA-PH-RWS-One-Pt Model. This model at first showed a gradual decrease of penalty value, then by generation 30 a sharp decrease of penalty occurred. The penalty value is then slowly decreasing until around generation 53, where the least penalty value was obtained with the best-so-far penalty was below 500. In the process of running until generation 200, some good penalty values were also obtained. However, the penalty values that were greater than the least penalty value obtained as in generation 53 do not appear in the graph as the inclusion of the elitism procedure keeps the least penalty value in each generation. The termination criterion in this research was based on user identified generation number; therefore the process was kept on running until generation 200 was reached.

The brown colour line shows a sample solution obtained from the EA-PH-QB-One-Pt Model, where it shows a sharp decrease of penalty value in some points around generation 2, 40, 120 and 132. The best-so-far penalty of about 500 was obtained around generation 195. On the other hand, the blue colour graph of EA-PH-RT-One-Pt Model shows only decreases of penalty values at the early generation. Around generation 14, the least penalty values were obtained where the best-so-far penalty was about 500. In the process of running until generation 200, some good penalty values were also obtained. However, the penalty values that were greater than the least penalty as in generation 14 do not appear in the graph as the inclusion of the elitism procedure keeps the best penalty value in each generation. Based on these three sample solution, the EA-PH-RWS-One-Pt give the least penalty value, while EA-PH-QB-One-Pt give the highest penalty.

### 6.3.2 Comparative evaluation of selection operators

This subsection compiles the results obtained from experimentations with three selection operators. They are the RWS, QB and RT Selections. The experimentations were run using the same parameters including the number of ingredients, type of ingredients, price of ingredients, ingredient weight, and EA parameters such as population size and generation number. All models were run 30 times each to produce a sample of statistically relevant result.

Table 6.8 shows the best-so-far solution for each model obtained. The simulated results of all EA models were compared based on their penalty values, that is, the best-so-far penalty, average penalty and standard deviation. Other results are in the forms of processing time (in seconds) and number of infeasible solutions. These five elements were used as indicators to evaluate the performance of these EA specialized models.

Table 6.8

*Comparative Results of Different Selection Operators*

| **Model** | **Best-so-far penalty** | **Average penalty** | **Standard deviation** | **Average run time (sec.)** | **Number of infeasible solution** |
|---|---|---|---|---|---|
| EA-PH-RWS-One-Pt | 300 | 547.0000 | 134.6298 | 12091.5000 | 0/30 |
| EA-PH-QB-One-Pt | 410 | 660.4762 | 141.3678 | 3079.5555 | 9/30 |
| EA-PH-RT-One-Pt | 340 | 700.3571 | 131.6978 | 3119.5000 | 2/30 |

The best-so-far solution obtained was for the EA-PH-RWS-One-Pt Model with a penalty value of 300, which was better than those of the EA-PH-QB-One-Pt Model and EA-PH-RT-One-Pt Model. In 30 runs of each set of generations, the best average penalty was achieved from the EA-PH-RWS-One-Pt Model. This is further supported by *z*-test to compare if the average of EA-PH-RWS-One-Pt with other models is significantly different. We found that, with confidence level of 95%, the difference between the EA-PH-RWS-One-Pt and the EA-PH-QB-One-Pt Model was significant with a *p*-value of 0.006. The difference between EA-PH-RWS-One-Pt and EA-PH-RT-One-Pt Model was also significant with a *p*-value of 0.000. Therefore, the performance of the EA-PH-RWS-One-Pt Model can be concluded as better than the others. The result of Shapiro-Wilk's test confirmed that the distributions from all models were normal.

In addition, the EA-PH-RWS-One-Pt Model produced no infeasible solution, while the performance of newly introduced EA-PH-RT-One-Pt Model is comparable, with only two infeasible values obtained. Meanwhile, the EA-PH-QB-One-Pt Model performs the poorest with nine infeasible solutions obtained.

The average times taken for the EA-PH-QB-One-Pt Model and EA-PH-RT-One-Pt Model was equally the same, but for EA-PH-RWS-One-Pt Model the time taken was approximately three times longer than those of the other two. In terms of standard deviation, all models gave almost the same value. Standard deviation reflects the deviation of a solution from the average value of each model. However, this is not an issue as users could choose the best solution in any run for implementation purposes.

Table 6.9 shows a sample solution for each of the EA model where seven ingredients were selected for the EA-PH-RWS-One-Pt Model, eight ingredients for the EA-PH-QB-One-Pt Model, and nine ingredients for the EA-PH-RT-One-Pt Model. The zero value means that the ingredient is not included in the combination of diet. Table 6.10 shows the composition of nutrient values in each corresponding solution. In every model, some soft constraints which are ingredients range and nutrients range were violated but all hard constraints were satisfied. This sample solution shows that the total cost for the ingredient mix of EA-PH-QB-One-Pt Model is the lowest with RM 134.29, whereas for EA-PH-RWS-One-Pt and EA-PH-RT-One-Pt obtained RM 238.13 and RM 220.08, respectively. However, since the total cost is greater, but with fewer constraints violated, the EA-PH-RWS-One-Pt is still more relevant to be chosen as the best solution.

Table 6.9

*Sample Solutions for Three Different EA Models*

| **Ingredient** | **Minimum (kg)** | **Maximum (kg)** | **EA-PH-RWS-One-Pt Model** | **EA-PH-QB-One-Pt Model** | **EA-PH-RT-One-Pt Model** |
|---|---|---|---|---|---|
| $X_1$ | 5 | 10 | 0 | 9.3987 | 10.3201 |
| $X_2$ | 15 | 50 | 0 | 43.2535 | 28.7322 |
| $X_3$ | 3 | 5 | 5.1456 | 4.8377 | 0 |
| $X_4$ | 5 | 50 | 30.0305 | 0 | 23.6823 |
| $X_5$ | 30 | 40 | 39.3687 | 0 | 19.8341 |
| $X_6$ | 5 | 15 | 0 | 11.0613 | 0 |
| $X_7$ | 5 | 15 | 0 | 0 | 11.7725 |
| $X_8$ | 2 | 5 | 0 | 0 | 0.0615 |
| $X_9$ | 15 | 60 | 0 | 0 | 0 |
| $X_{10}$ | 5 | 15 | 0 | 12.3407 | 0 |
| $X_{11}$ | 5 | 15 | 11.5438 | 10.6669 | 0 |
| $X_{12}$ | 3 | 5 | 5.3695 | 4.3561 | 0.4605 |
| $X_{13}$ | 3 | 5 | 3.4875 | 0 | 4.3874 |
| $X_{14}$ | 3 | 5 | 5.4861 | 4.3561 | 0.3427 |
| *Total weight (kg)* | | | 100.4317 | 100.4952 | 99.5933 |
| *Total cost (RM)* | | | 238.13 | 134.29 | 220.08 |

Table 6.10

*Nutrients Values for different EA Model Solutions*

| Nutrients | Minimum | Maximum | EA-PH-RWS-One-Pt Model | EA-PH-QB-One-Pt Model | EA-PH-RT-One-Pt Model |
|---|---|---|---|---|---|
| Crude Protein, % | 38.00 | 45.00 | 41.5328 | 43.2310 | 41.5534 |
| Lipid, % | 0.08 | 0.18 | 5.7381 | 4.4771 | 5.8115 |
| Fibre, % | 0 | 4.00 | 3.5017 | 6.0078 | 4.7898 |
| Ash, % | 0 | 15.0 | 10.2398 | 9.4372 | 10.4196 |
| Calcium, % | 0 | 2.30 | 1.9881 | 1.8679 | 1.3983 |
| Phosphorus, % | 0.30 | 0.70 | 1.1629 | 1.1469 | 1.0050 |
| Arginine, % | 2.2 | 2.32 | 2.1570 | 2.6589 | 2.7407 |
| Histidine, % | 0.6 | 0.84 | 0.9648 | 1.0541 | 1.0184 |
| Isoleucine, % | 1.0 | 1.33 | 5.7968 | 5.0654 | 2.2921 |
| Leucine, % | 1.7 | 2.16 | 7.3826 | 6.8225 | 3.6547 |
| Lysine, % | 1.55 | 1.65 | 2.4929 | 2.5920 | 2.8756 |
| Methionine, % | 0.7 | 0.96 | 0.8836 | 0.5765 | 0.9273 |
| Phenylalanine, % | 1.4 | 1.6 | 1.6745 | 2.0093 | 1.9787 |
| Threonine, % | 1.3 | 1.44 | 0.3844 | 0.4175 | 0.4533 |
| Tryptophan, % | 0.2 | 0.32 | 1.4556 | 1.5453 | 1.7018 |
| Valine | 1.4 | 1.6 | 2.0589 | 2.1667 | 2.2159 |
| Methionine + Cystine, % | 1.0 | 1.44 | 1.3544 | 1.1757 | 1.4684 |
| Phenylalanine + Tyrosine, % | 2.7 | 7.1 | 2.6158 | 3.0639 | 3.1231 |
| Calcium:Phosphorus, % | 0.7692 | 0.7692 | 1.7096 | 1.6287 | 1.3913 |

As a whole, the EA-PH-RWS-One-Pt Model with Roulette Wheel Selection operator is still the best selection operator compared to the other two operators, especially in terms of the best-so-far penalty. The performance of the newly introduced EA-PH-RT-One-Pt Model with Roulette-Tournament Selection operator is comparable with EA-PH-RWS-One-Pt in terms of the least number of infeasible solutions obtained.

## 6.4 Experimentations with different crossover operators

Experimentations of EA with two crossover operators that are the One-Point Crossover and Average Crossover were conducted. While these crossovers were each tested, the other operators remained the same. These controlled operators were initialization with incorporation of Power Heuristics, Roulette Wheel Selection, and Power Mutation. The parameter values used in this experimentation is the same as in Table 6.2. Figure 6.3 shows the process flow of the evolutionary cycle for the steps employed in both models. Discussion on the performance of these two variations of EA model can be referred to in Subsection 6.4.1. Then, the overall comparison on the performance of these crossover operators can be referred to in Subsection 6.4.2.

### 6.4.1 The process flows of different crossover operators

The generic process flow is shown in the Figure 6.3.

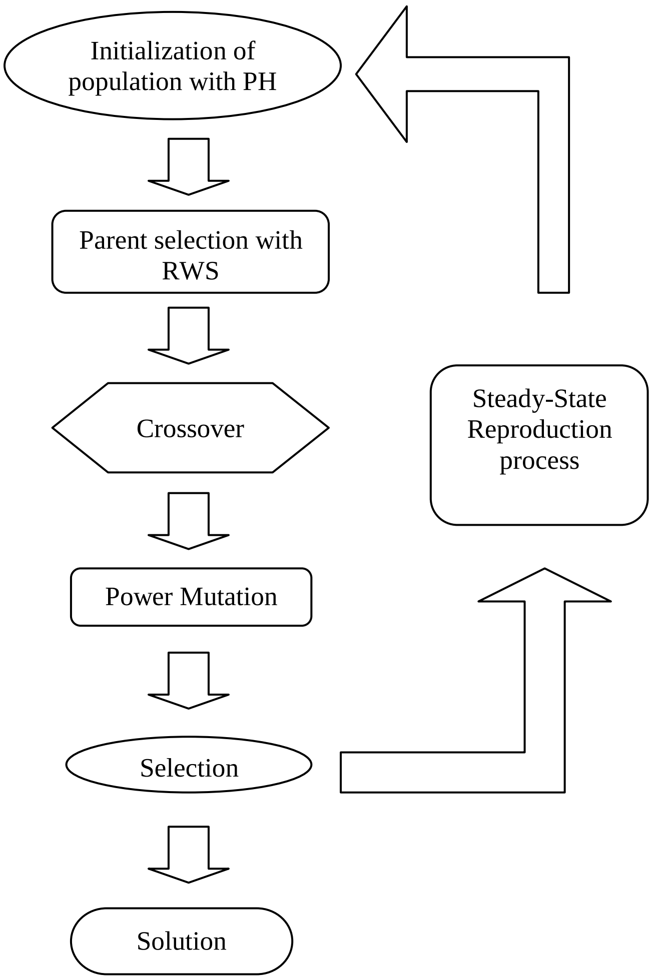



*Figure 6.3.* The Process Flow of EA Model

Initially, the One-Point Crossover was employed when the EA-PH-RWS-One-Point Model was tested. In the experimentation the One-Point Crossover operator was later replaced by the Average Crossover operator. This altered model is known as the EA-PH-RWS-Avg Model. The proposed Average Crossover is developed to disrupt the chromosome more in a manner of getting the average value between two chromosomes. Figure 6.4 shows the graph obtained on a sample solution for each of these models.

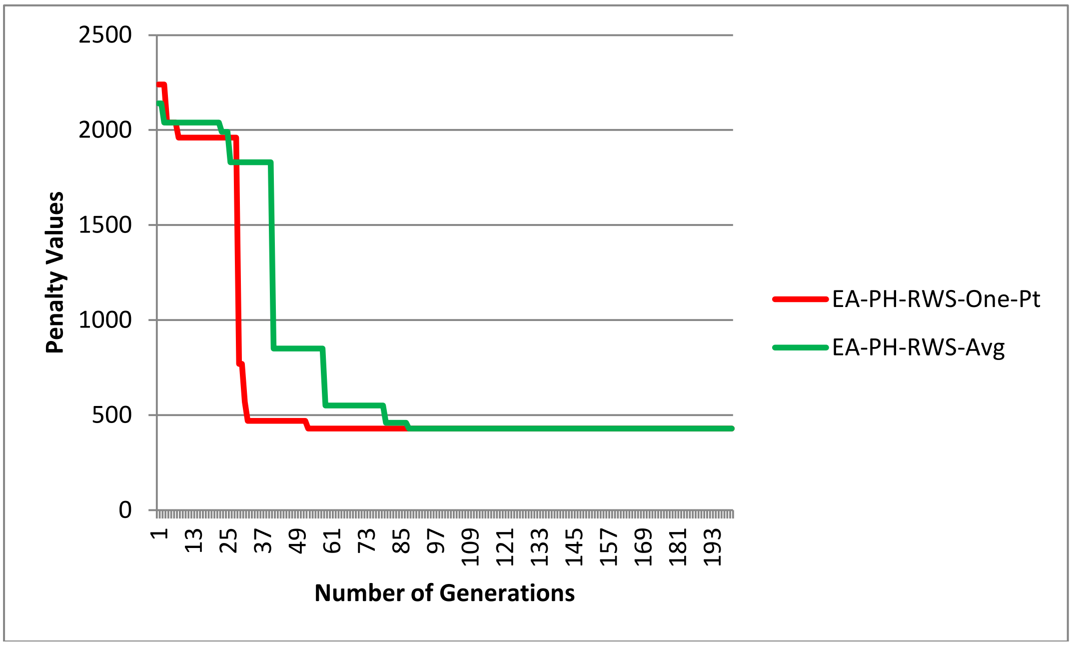



*Figure 6.4.* Sample Solutions of the Two Models with Different Crossover Operators

Figure 6.4 exhibits a sample solution obtained from each EA-PH-RWS-One-Pt and EA-PH-RWS-Avg Models. The sample solution for the EA-PH-RWS-One-Pt is the same as the previous graph in Subsection 6.3, where the best-so-far penalty of less than 500 was obtained around generation 53.

The green coloured graph of EA-PH-RWS-Avg Model shows several points of penalty value in a decreasing manner. Around generation 90, the least penalty values were obtained where the best-so-far penalty value was below 500. The termination criterion in this research was the user identified generation number; therefore the process was kept on running until generation 200 was reached.

Both models give the same penalty values at the end of the running process. The least penalty value for both models was obtained in less than 100 generations. From these sample solution, we can say that the performance of both models are identical.

### 6.4.2 Comparative evaluation of crossover operators

We applied two different EA models using real diet formulation problem. These two procedures were:

1. EA-PH-RWS-One-Pt Model: The EA with One-Point Crossover
2. EA-PH-RWS-Avg Model: The EA with Average Crossover

RWS was utilized in these EA models due to its good performance. These two models were run based on the parameter values given in Table 6.2. Each EA model with each crossover operator was run for 30 sets to provide a sample of statistically relevant results. Results obtained from the experimentations are shown in Table 6.11, which consists of five elements: best-so-far penalty, average penalty, standard deviation, average run time, and number of infeasible solution.

Table 6.11

*Comparative Results of Different Crossover Operators*

| **Model** | **Best-so-far penalty** | **Average penalty** | **Standard deviation** | **Average run time (sec.)** | **Number of infeasible solution** |
|---|---|---|---|---|---|
| EA-PH-RWS-One-Pt Model | 300 | 547.0000 | 134.6298 | 12091.5000 | 0/30 |
| EA-PH-RWS-Avg Model | 300 | 520.6667 | 126.5166 | 12101.4500 | 0/30 |

Based on the results in Table 6.11, an equally same best-so-far solution for EA models with penalty of 300 was obtained. Among the two models, the best average penalty was achieved from the EA-PH-RWS-Avg Model. z-test was conducted to compare if the average penalty of the EA-PH-RWS-Avg Model is significantly different from that of the EA-PH-RWS-One-Pt Model. The results obtained from the test showed that with 95% confidence, the difference between both models was not statistically significant. It means that the performance of both models is identical. The result from Shapiro-Wilk's test confirmed that the distributions from both models were normal.

Even though the mean or average is not significantly different from the other one, The EA-PH-RWS-Avg Model is also able to produce the best-so-far solutions with the least standard deviation as compared to that of the EA-PH-RWS-One-Pt Model. It means that the diversity of the solution penalty values is in smaller range and closer to its average penalty value. Statistically, the smaller the range, the better is the solution.

In terms of generating infeasible solution, both the EA-PH-RWS-Avg and EA-PH-RWS-One-Pt Models are able to produce no infeasible solutions for all 30 sets.

Besides, in terms of processing time, both models require approximately the same length of time, which is about 200 minutes.

Table 6.12 shows a sample solution for each EA model where seven ingredients were selected for the EA-PH-RWS-One-Pt Model and six ingredients for the EA-PH-RWS-Avg Model. The zero value means that the ingredient is not included in the combination of diet. Table 6.13 shows the composition of nutrient values in each solution. In every model, some soft constraints which are ingredients range and nutrients range were violated; however all hard constraints were satisfied. The total cost of ingredients mix for EA-PH-RWS-Avg Model was RM 175.86, which was approximately RM 62 lower than that total cost obtained for EA-PH-RWS-One-Pt Model.

Table 6.12

*Sample Solutions of Different EA Models*

| **Ingredient** | **Minimum (kg)** | **Maximum (kg)** | **EA-PH-RWS-One-Pt Model** | **EA-PH-RWS-Avg-Pt Model** |
|---|---|---|---|---|
| Rice bran, Malaysia, $X_1$ | 5 | 10 | 0 | 0 |
| Soybean meal, $X_2$ | 15 | 50 | 0 | 0 |
| Palm kernel cake, $X_3$ | 3 | 5 | 5.1456 | 4.9589 |
| Local fishmeal, $X_4$ | 5 | 50 | 30.0305 | 36.0149 |
| Wheat flour, $X_5$ | 30 | 40 | 39.3687 | 38.7124 |
| Wheat pollard, $X_6$ | 5 | 15 | 0 | 0 |
| Poultry meal, $X_7$ | 5 | 15 | 0 | 0 |
| Crude Palm Oil, $X_8$ | 2 | 5 | 0 | 3.2949 |
| Imported fish meal, $X_9$ | 15 | 60 | 0 | 0 |
| Meat and bone meal, $X_{10}$ | 5 | 15 | 0 | 0 |
| Poultry by product, $X_{11}$ | 5 | 15 | 11.5438 | 13.5839 |
| Blood meal, $X_{12}$ | 3 | 5 | 5.3695 | 0 |
| Krill meal, $X_{13}$ | 3 | 5 | 3.4875 | 0 |
| Squid meal, $X_{14}$ | 3 | 5 | 5.4861 | 3.8604 |
| *Total weight (kg)* | | | 100.4317 | 100.4254 |
| *Total cost (RM)* | | | 238.13 | 175.86 |

Table 6.13

*Nutrients Values for the different EA Model Solutions*

| Nutrients | Minimum | Maximum | EA-PH-RWS-One-Pt Model | EA-PH-RWS-Avg Model |
|---|---|---|---|---|
| Crude Protein, % | 38.00 | 45.00 | 41.5328 | 38.0912 |
| Lipid, % | 0.08 | 0.18 | 5.7381 | 9.1064 |
| Fibre, % | 0 | 4.00 | 3.5017 | 3.6863 |
| Ash, % | 0 | 15.0 | 10.2398 | 9.7297 |
| Calcium, % | 0 | 2.30 | 1.9881 | 2.3540 |
| Phosphorus, % | 0.30 | 0.70 | 1.1629 | 1.3591 |
| Arginine, % | 2.2 | 2.32 | 2.1570 | 2.0645 |
| Histidine, % | 0.6 | 0.84 | 0.9648 | 0.7657 |
| Isoleucine, % | 1.0 | 1.33 | 5.7968 | 4.5444 |
| Leucine, % | 1.7 | 2.16 | 7.3826 | 5.5968 |
| Lysine, % | 1.55 | 1.65 | 2.4929 | 2.2178 |
| Methionine, % | 0.7 | 0.96 | 0.8836 | 0.8534 |
| Phenylalanine, % | 1.4 | 1.6 | 1.6745 | 1.3858 |
| Threonine, % | 1.3 | 1.44 | 0.3844 | 0.3430 |
| Tryptophan, % | 0.2 | 0.32 | 1.4556 | 1.2839 |
| Valine | 1.4 | 1.6 | 2.0589 | 1.7612 |
| Methionine + Cystine, % | 1.0 | 1.44 | 1.3544 | 1.3044 |
| Phenylalanine + Tyrosine, % | 2.7 | 7.1 | 2.6158 | 2.3495 |
| Calcium:Phosphorus, % | 0.7692 | 0.7692 | 1.7096 | 1.7320 |

In conclusion, our newly introduced EA-PH-RWS-Avg Model could produce a comparable solution with the EA-PH-RWS-One-Pt Model. Therefore, the Average Crossover operator could be utilized in any real world problems that have a real value representation.

## 6.5 Overall performance of EA model variants

Subsequent operator in our EA cycle is the mutation operator. In this research, we only introduce a specialized mutation operator, namely the Power Mutation. This mutation has been incorporated in all previous experimentations, thus our EA experiment is completed. Hence, in this section, we evaluate the overall performance of the three main EA model with their respective model variants. These models are:

i. EA with PH initialization, RWS, Power Mutation and two specialized crossover operators:

a. EA-PH-RWS-Avg

b. EA-PH-RWS-One-Pt

ii. EA with PH initialization, QB Selection, Power Mutation and two specialized crossover operators

a. EA-PH-QB-One-Pt

b. EA-PH-QB-Avg

iii. EA with PH initialization, RT Selection, Power Mutation and two specialized crossover operators

a. EA-PH-RT-One-Pt

b. EA-PH-RT-Avg

The performance evaluation on these six model variants is to identify the best combination in terms of selection and crossover operators. As in previous comparisons, the same data and EA parameters were used. Results of these evaluations can be viewed in Table 6.14. Again, the elements being compared are best-so-far penalty, average penalty, standard deviation, average run time, and number of infeasible solutions.

Table 6.14

*Comparative Results of Different EA Model Variants*

| **Model / Sub-model** | **Best-so-far penalty** | **Average penalty** | **Standard deviation** | **Average run time (sec.)** | **Number of infeasible solution** |
|---|---|---|---|---|---|
| EA-PH-RWS-Avg | 300 | 520.6667 | 126.5166 | 12101.4500 | 0/30 |
| EA-PH-RWS-One-Pt | 300 | 547.0000 | 134.6298 | 12091.5000 | 0/30 |
| EA-PH-QB-One-Pt | 410 | 660.4762 | 141.3675 | 3079.5555 | 9/30 |
| EA-PH-QB-Avg | 340 | 615.2000 | 159.0367 | 3458.5000 | 5/30 |
| EA-PH-RT-One-Pt | 340 | 700.3571 | 131.6978 | 3119.5000 | 2/30 |
| EA-PH-RT-Avg | 370 | 616.7857 | 118.4797 | 3139.1111 | 2/30 |

The EA-PH-RWS-One-Pt and the EA-PH-RWS-Avg produced the same best-so-far solution in terms of penalty value, that is 300. Subsequently, the EA-PH-QB-Avg and EA-PH-RT-One-Pt gave an acceptable good solution with penalty value of 340. On the other hand, the EA-PH-RT-Avg gave a penalty value of 370, while the EA-PH-QB-One-Pt gave the worst best-so-far penalty value of 410.

In terms of the average penalty value, the EA-PH-RWS-Avg provides the best value followed by the EA-PH-RWS-One-Pt, EA-PH-QB-Avg, and then the EA-PH-RT-Avg. The worst penalty value is from the EA-PH-QB-One-Pt and the EA-PH-RT-One-Pt. Analysis of variance (ANOVA) test was conducted to compare if the average penalty of the EA-PH-RWS-Avg is significantly different from the other models. With 95% confidence, we found that the mean difference between the EA-PH-RWS-Avg and the EA-PH-QB-One-Pt are significant with *p*-values of 0.006. The mean difference between the EA-PH-RWS-Avg and the EA-PH-RT-One-Pt was also significant with *p*-values of 0.000. Therefore, we can conclude that the performance of the EA-PH-RWS-Avg is better than the EA-PH-QB-One-Pt and EA-PH-RT-One-Pt. However, the performance of the EA-PH-RWS-Avg with other models that are EA-PH-RWS-One-Pt, EA-PH-QB-Avg and EA-PH-RT-Avg is identical. Normality test was done to check whether the data are normally distributed or not. Result from Shapiro-Wilk's test confirmed that all distributions from each model variant normal. The assumption of normality is a prerequisite for many statistical tests including the ANOVA. Therefore, ANOVA test conducted was statistically relevant.

The standard deviation for the EA-PH-RT-Avg was the lowest, followed by the EA-PH-RWS-Average, EA-PH-RT-One-Pt and the EA-PH-RWS-One-Pt. The model variants with Queen-Bee Selection gave the biggest value either when One-Point Crossover (EA-PH-QB-One-Pt) or Average Crossover (EA-PH-QB-Avg) was used. Standard deviation shows the deviation or dispersion of the data from its mean. A low standard deviation indicates that the model variant gives a stable solution which is always close to its average penalty value.

In terms of the number of infeasible solutions generated, the EA-PH-RWS-Avg and the EA-PH-RWS-One-Pt are still the best compared to any other variants. Again, both the EA-PH-QB-One-Pt and EA-PH-QB-Avg gave the largest number of infeasible solutions with nine and five, respectively, from the total number of runs. Both model variants with Roulette-Tournament Selection produced rather an equal number of infeasible solutions which is also can be considered as good performance.

However, in terms of processing time, both EA-PH-RWS-Avg and EA-PH-RWS-One-Pt Models require approximately the same length of time, which is about 200 minutes. Other four models which are EA-PH-QB-One-Pt, EA-PH-QB-Avg, EA-PH-RT-One-Pt and EA-PH-RT-Avg require faster time which is approximately 53 minutes.

Overall, the EA-PH-RWS-Avg provides the least penalty value, the best average penalty value, low standard deviation also with no infeasible solutions. In terms of run time, the model variants that employed the RWS procedure obviously required more run times than those employed other selection procedures. It is about 200 minutes.

Subsequently, Table 6.15 shows a sample solution for each of the six EA model variants where seven ingredients were selected for the EA-PH-RWS-One-Pt, six ingredients for the EA-PH-RWS-Avg, eight ingredients for the EA-PH-QB-One-Pt, eight ingredients for the EA-PH-QB-Avg, nine ingredients for the EA-PH-RT-One-Pt, and seven ingredients for the EA-PH-RT-Avg. The ingredients mix varies from six to nine to provide the most appropriate feed mix for the shrimp as outlined by all relevant requirements. Zero value means that the ingredient is not included in the combination of diet.

Based on this sample solution, the least cost is obtained from EA-PH-QB-One-Pt model variant with RM 134.29. EA-PH-RT-Avg and EA-PH-QB-Avg also gave slightly the same cost which is of RM 134.39 and RM 140.86, respectively. However, the EA-PH-RWS-Avg model variant requires RM 175.86 which is considered good enough.  The total cost obtained from the other two model variants is quite high where the different of approximately RM 100 with that of the EA-PH-QB-One-Pt model variant. Consequently, in this case, the solution with the least penalty value is still a priority to be chosen. Therefore, the solution for EA-PH-RWS-Avg is the best with less penalty value and appropriate total cost of RM 175.86.

Subsequently, Table 6.16 shows the composition of nutrient values in each solution. In every model variant, some ingredients range and nutrients range constraint in the soft category were violated, but all hard constraints were satisfied.

Overall, based on the three main EA models that we experimented in this research the EA-PH-RWS-Avg variant has a better solution except for the run time. However, this is not a big matter since computer facilities and technologies develop very rapidly which could be of help later. In fact, 85.7% of the feasible solutions were obtained in the first 100 generations. Therefore, for the subsequent analyses, the EA-PH-RWS-Avg model variant is experimental further with different scenarios and is described in the next sections.

Table 6.15

*A Sample Solution of Different EA Models*

| **Ingredient** | **Minimum (kg)** | **Maximum (kg)** | **EA-PH-RWS-One-Pt Model** | **EA-PH-RWS-Avg-Pt Model** | **EA-PH-QB-One-Pt Model** | **EA-PH-QB-Avg Model** | **EA-PH-RT-One-Pt Model** | **EA-PH-RT-Avg Model** |
|---|---|---|---|---|---|---|---|---|
| Rice bran, Malaysia, $X_1$ | 5 | 10 | 0 | 0 | 9.3987 | 8.9442 | 10.3201 | 10.3016 |
| Soybean meal, $X_2$ | 15 | 50 | 0 | 0 | 43.2535 | 45.4986 | 28.7322 | 49.0668 |
| Palm kernel cake, $X_3$ | 3 | 5 | 5.1456 | 4.9589 | 4.8377 | 3.9131 | 0 | 0 |
| Local fishmeal, $X_4$ | 5 | 50 | 30.0305 | 36.0149 | 0 | 0 | 23.6823 | 0 |
| Wheat flour, $X_5$ | 30 | 40 | 39.3687 | 38.7124 | 0 | 0 | 19.8341 | 0 |
| Wheat pollard, $X_6$ | 5 | 15 | 0 | 0 | 11.0613 | 11.7874 | 0 | 13.0532 |
| Poultry meal, $X_7$ | 5 | 15 | 0 | 0 | 0 | 12.205 | 11.7725 | 11.7961 |
| Crude Palm Oil, $X_8$ | 2 | 5 | 0 | 3.2949 | 0 | 3.2124 | 0.0615 | 4.4926 |
| Imported fish meal, $X_9$ | 15 | 60 | 0 | 0 | 0 | 0 | 0 | 0 |
| Meat and bone meal, $X_{10}$ | 5 | 15 | 0 | 0 | 12.3407 | 11.2183 | 0 | 0 |
| Poultry by product, $X_{11}$ | 5 | 15 | 11.5438 | 13.5839 | 10.6669 | 0 | 0 | 0 |
| Blood meal, $X_{12}$ | 3 | 5 | 5.3695 | 0 | 4.3561 | 0 | 0.4605 | 5.8867 |
| Krill meal, $X_{13}$ | 3 | 5 | 3.4875 | 0 | 0 | 3.5733 | 4.3874 | 0 |
| Squid meal, $X_{14}$ | 3 | 5 | 5.4861 | 3.8604 | 4.3561 | 0 | 0.3427 | 4.8233 |
| *Total weight (kg)* | | | 100.4317 | 100.4254 | 100.4952 | 100.3523 | 99.5933 | 99.4203 |
| *Total cost (RM)* | | | 238.13 | 175.86 | 134.29 | 140.86 | 220.08 | 134.39 |

Table 6.16

*Nutrients Value for EA Models Solution*

| **Nutrients** | **Minimum** | **Maximum** | **EA-PH-RWS-One-Pt Model** | **EA-PH-RWS-Avg Model** | **EA-PH-QB-One-Pt Model** | **EA-PH-QB-Avg Model** | **EA-PH-RT-One-Pt Model** | **EA-PH-RT-Avg Model** |
|---|---|---|---|---|---|---|---|---|
| Crude Protein, % | 38.00 | 45.00 | 41.5328 | 38.0912 | 43.2310 | 39.4529 | 41.5534 | 41.0608 |
| Lipid, % | 0.08 | 0.18 | 5.7381 | 9.1064 | 4.4771 | 7.9945 | 5.8115 | 7.5929 |
| Fibre, % | 0 | 4.00 | 3.5017 | 3.6863 | 6.0078 | 5.8065 | 4.7898 | 5.1246 |
| Ash, % | 0 | 15.0 | 10.2398 | 9.7297 | 9.4372 | 9.7813 | 10.4196 | 6.8319 |
| Calcium, % | 0 | 2.30 | 1.9881 | 2.3540 | 1.8679 | 1.4066 | 1.3983 | 0.2852 |
| Phosphorus, % | 0.30 | 0.70 | 1.1629 | 1.3591 | 1.1469 | 0.9682 | 1.0050 | 0.4452 |
| Arginine, % | 2.2 | 2.32 | 2.1570 | 2.0645 | 2.6589 | 2.7778 | 2.7407 | 2.5113 |
| Histidine, % | 0.6 | 0.84 | 0.9648 | 0.7657 | 1.0541 | 0.9583 | 1.0184 | 1.0809 |
| Isoleucine, % | 1.0 | 1.33 | 5.7968 | 4.5444 | 5.0654 | 1.8899 | 2.2921 | 5.4071 |
| Leucine, % | 1.7 | 2.16 | 7.3826 | 5.5968 | 6.8225 | 3.2322 | 3.6547 | 7.2491 |
| Lysine, % | 1.55 | 1.65 | 2.4929 | 2.2178 | 2.5920 | 2.5970 | 2.8756 | 2.5496 |
| Methionine, % | 0.7 | 0.96 | 0.8836 | 0.8534 | 0.5765 | 0.6561 | 0.9273 | 0.5581 |
| Phenylalanine, % | 1.4 | 1.6 | 1.6745 | 1.3858 | 2.0093 | 1.9486 | 1.9787 | 2.0632 |
| Threonine, % | 1.3 | 1.44 | 0.3844 | 0.3430 | 0.4175 | 0.4184 | 0.4533 | 0.4382 |
| Tryptophan, % | 0.2 | 0.32 | 1.4556 | 1.2839 | 1.5453 | 1.5785 | 1.7018 | 1.5441 |
| Valine | 1.4 | 1.6 | 2.0589 | 1.7612 | 2.1667 | 2.0648 | 2.2159 | 2.1325 |
| Methionine + Cystine, % | 1.0 | 1.44 | 1.3544 | 1.3044 | 1.1757 | 0.9901 | 1.4684 | 1.1678 |
| Phenylalanine + Tyrosine, % | 2.7 | 7.1 | 2.6158 | 2.3495 | 3.0639 | 0.5581 | 3.1231 | 3.1515 |
| Calcium:Phosphorus, % | 0.7692 | 0.7692 | 1.7096 | 1.7320 | 1.6287 | 1.2143 | 1.3913 | 0.6406 |

### 6.6 What-if analysis

What-if analysis was conducted as a part of comparative evaluation on the solutions. This analysis considers a 'what if' question that will lead to what can happen when some variables are changed. In this analysis, two scenarios were considered, that are, when the total ingredient weight is increased and the price of ingredients decreases. Hence, in this section two EA models are evaluated based on the proposed EA-PH-RWS-Avg to see the impact on the changes of the parameters obtained including the penalty value and processing time. The two scenarios are:

Weight Increased: What if the total ingredient weight is increased by 500 kg and the price of each ingredient remains?

Price Decreased: What if the price of each ingredient is decreased by RM 0.30 and the total ingredient weight remains?

#### 6.6.1 Experimentation with Weight Increased scenario

The Weight Increased scenario describes the situation where the total ingredient weight is increased by 500 kg and the price of each ingredient remains the same. Practitioners especially farmers tend to buy more than 100 kg of feed for shrimp depending on their farm sizes. In this case, 500 kg is reasonable to be considered for evaluation purposes. Table 6.17 shows the analysis for this scenario, where the best so far penalty value obtained is 330. The average penalty value is raised to 602.0000 and the standard deviation is increased to 146.3481. On average, the run time is about the same which is about 196 minutes. In 10 runs, there is no infeasible solution obtained.

Table 6.17

*Analysis of Weight Increased Scenario*

| **Procedure** | **Best-so-far penalty** | **Average penalty** | **Standard deviation** | **Average run time (sec.)** | **Number of infeasible solution** |
|---|---|---|---|---|---|
| EA-PH-RWS-Avg Model | 330 | 602.0000 | 146.3481 | 11787.9500 | 0/10 |

Subsequently, Table 6.18 represents a sample solution for the EA-PH-RWS-Avg with a list of recommended ingredients obtained from the experimentation. The minimum value of each ingredient was either zero or as stated in Table 6.18. The total selected ingredients were nine where all the ingredients lay in the permitted range.

Table 6.18

*A Sample Solution of Weight Increased Scenario*

| **Ingredient** | **Minimum (kg)** | **Maximum (kg)** | **Quantity (kg)** |
|---|---|---|---|
| Rice bran, Malaysia, $X_1$ | 25 | 50 | 26.2351 |
| Soybean meal, $X_2$ | 75 | 250 | 93.3251 |
| Palm kernel cake, $X_3$ | 15 | 25 | 15.2145 |
| Local fishmeal, $X_4$ | 25 | 250 | 37.2142 |
| Wheat flour, $X_5$ | 150 | 200 | 150.0241 |
| Poultry meal, $X_7$ | 25 | 75 | 63.2062 |
| Crude Palm Oil, $X_8$ | 10 | 25 | 12.5061 |
| Imported fish meal, $X_9$ | 75 | 300 | 80.0807 |
| Krill meal, $X_{13}$ | 15 | 25 | 22.2714 |
| *Total weight (kg)* | | | 500.0774 |
| *Total cost (RM)* | | | 1173.504 |

Table 6.19 shows the nutrients value for the sample of solution. Some nutrients are in the permitted range and other nutrients are violated. However, this solution is feasible as it satisfies all hard constraints. The total ingredient weight for this solution is 500.0774 kg and the total cost is RM1173.504. Since the total ingredient weight is five times the original value, the total cost is also approximately five times

more. Therefore, we can say the performance of this model variant is stable and it can adopt changes in the total ingredient weight.

Table 6.19

*Nutrients Value for Weight Increased Scenario*

| Nutrients | Minimum | Maximum | Quantity |
|---|---|---|---|
| Crude Protein, % | 38.00 | 45.00 | 39.2859 |
| Lipid, % | 0.08 | 0.18 | 8.1198 |
| Fibre, % | 0 | 4.00 | 4.3493 |
| Ash, % | 0 | 15.0 | 9.3399 |
| Calcium, % | 0 | 2.30 | 1.1138 |
| Phosphorus, % | 0.30 | 0.70 | 0.8633 |
| Arginine, % | 2.2 | 2.32 | 2.4218 |
| Histidine, % | 0.6 | 0.84 | 0.9149 |
| Isoleucine, % | 1.0 | 1.33 | 2.5149 |
| Leucine, % | 1.7 | 2.16 | 3.7492 |
| Lysine, % | 1.55 | 1.65 | 2.5736 |
| Methionine, % | 0.7 | 0.96 | 0.9077 |
| Phenylalanine, % | 1.4 | 1.6 | 1.7744 |
| Threonine, % | 1.3 | 1.44 | 0.4162 |
| Tryptophan, % | 0.2 | 0.32 | 1.5500 |
| Valine, % | 1.4 | 1.6 | 2.0150 |
| Methionine + Cystine, % | 1.0 | 1.44 | 7.0133 |
| Phenylalanine + Tyrosine, % | 2.7 | 7.1 | 2.8097 |
| Calcium:Phosphorus, % | 0.7692 | 0.7692 | 1.2902 |

### 6.6.2 Experimentation of Price Decreased scenario

The Price Decreased scenario represents the situation where the price of each ingredient is decreased by RM 0.30 and the total ingredient weight remains at 100 kg. The price of the ingredients always fluctuates depending on availability, demands and currency exchange. This scenario can happen when many ingredients are available when demand is low. It may also happen when Malaysian Ringgit (MYR) increases its value. Table 6.20 presents the analysis obtained from the experimentation and Table 6.21 shows the solution, that is, the quantity of each ingredient produced. The nutrient values obtained for the solution is depicted in Table 6.22.

The average penalty value is equally the same as the average penalty value obtained from the EA-PH-RWS-Avg model variant in section 6.5 with the value of 520.0000. A low standard deviation of 109.8484 is obtained which means that the performance of this model is stable, where the penalty value obtained in each run is always close to its average penalty value. The average run time is also approximately the same with that in the previous experimentation in section 6.5 with the value of 12181.1024. This is about 203 minutes. A higher best-so-far penalty is obtained with the value of 340 as compared to that in the previous experimentation; and in 10 runs, there is no infeasible solution obtained in this experiment as well.

Table 6.20

*Analysis for Decreased-Price Model*

| **Model** | **Best-so-far penalty** | **Average penalty** | **Standard deviation** | **Average run time (sec.)** | **Number of infeasible solution** |
|---|---|---|---|---|---|
| EA-PH-RWS-Avg Model | 340 | 520.0000 | 109.8484 | 12181.1024 | 0/10 |

From Table 6.21, the total ingredient weight obtained is 100.0496 kg, which has successfully achieved the suggested weight and the total price for this feed is RM 183.59. Ten ingredients were selected for this sample solution with four ingredients was out of the range, which are ingredients 5, 12, 13 and 14. These are actually wheat flour, blood meal, krill meal and squid meal. Other ingredients are in the permitted range. Some nutrients in Table 6.22 are fulfilled, and some are violated. However, these nutrients are classified as soft constraint whereby the impact is not critical if it is violated.

Table 6.21

*A Sample Solution of Price Decrease Scenario*

| Ingredient | Minimum (kg) | Maximum (kg) | Quantity (kg) |
|---|---|---|---|
| Palm kernel cake, $X_3$ | 3 | 5 | 3.9901 |
| Local fishmeal, $X_4$ | 5 | 50 | 28.1786 |
| Wheat flour, $X_5$ | 30 | 40 | 22.0504 |
| Poultry meal, $X_7$ | 5 | 15 | 10.1219 |
| Crude Palm Oil, $X_8$ | 2 | 5 | 3.6080 |
| Meat and bone meal, $X_{10}$ | 5 | 15 | 11.2993 |
| Poultry by product, $X_{11}$ | 5 | 15 | 13.1852 |
| Blood meal, $X_{12}$ | 3 | 5 | 0.512 |
| Krill meal, $X_{13}$ | 3 | 5 | 1.3918 |
| Squid meal, $X_{14}$ | 3 | 5 | 5.7073 |
| *Total weight (kg)* | | | 100.0496 |
| *Total cost (RM)* | | | 183.59 |

Table 6.22

*Nutrients Value for Decreased-Price Model Solution*

| Nutrients | Minimum | Maximum | Quantity |
|---|---|---|---|
| Crude Protein, % | 38.00 | 45.00 | 44.9431 |
| Lipid, % | 0.08 | 0.18 | 11.1914 |
| Fibre, % | 0 | 4.00 | 3.9766 |
| Ash, % | 0 | 15.0 | 13.3114 |
| Calcium, % | 0 | 2.30 | 3.1210 |
| Phosphorus, % | 0.30 | 0.70 | 1.7406 |
| Arginine, % | 2.2 | 2.32 | 2.5591 |
| Histidine, % | 0.6 | 0.84 | 0.8682 |
| Isoleucine, % | 1.0 | 1.33 | 6.1540 |
| Leucine, % | 1.7 | 2.16 | 7.4608 |
| Lysine, % | 1.55 | 1.65 | 2.5484 |
| Methionine, % | 0.7 | 0.96 | 0.9093 |
| Phenylalanine, % | 1.4 | 1.6 | 1.5541 |
| Threonine, % | 1.3 | 1.44 | 0.3687 |
| Tryptophan, % | 0.2 | 0.32 | 1.4912 |
| Valine | 1.4 | 1.6 | 2.0863 |
| Methionine + Cystine, % | 1.0 | 1.44 | 1.420 |
| Phenylalanine + Tyrosine, % | 2.7 | 7.1 | 2.5019 |
| Calcium:Phosphorus, % | 0.7692 | 0.7692 | 1.7928 |

## 6.7 Conclusion

In this chapter we have discussed the results obtained from several experiments that have been conducted. Evaluation and comparisons were made with regard to the selection and crossover operators. Based on experimentations with the selection operator, Roulette Wheel is the best. On the other hand, in experimentations for crossover operator, the Average Crossover produces slightly better solutions for most of the evaluation parameters such as best-so-far penalty, least average penalty value, and none infeasible solutions was obtained.

Evaluation and comparisons among the EA models were also carried out to observe the performance of each EA model in terms of the evaluation parameters used. The models are EA-PH-RWS-Avg, EA-PH-RWS-One-Pt, EA-PH-QB-One-Pt, EA-PH-QB-Avg, EA-PH-RT-One-Pt and EA-PH-RT-Avg. Results show that the EA-PH-RWS-Avg Model provides the best solution in most of the evaluation parameters except for duration of run time. The newly introduced EA-PH-RT-One-Pt and EA-PH-RT-Avg also produce acceptable good solutions when compared to the other EA models. However, models with Queen-Bee Selection especially the EA-PH-QB-One-Pt model variant produces quite a number of infeasible solutions, thus it is considered a poorly performed EA model.

In this research, we attempt to compile nutrients requirement data from various sources, even though the specific data still does not exist. However, the compilation of data results in a very small acceptable range of nutrients requirement, which is very hard to fulfil. Therefore, some relaxation of nutrients requirement is acceptable such as reported by Lara and Romero (1993) and Mitani and Nakayama (1997). Furuya et al. (1997) and most of the commercial feed companies carried out a strategy in which no minimum requirement is put as a limitation so that better solution with less penalty value is produced.

Further investigation on the proposed EA-PH-RWS-Avg was carried out with "what-if" analysis on two different scenarios. The result obtained from the experimentation shows that the performance of our newly proposed EA-PH-RWS-Avg can adopt changes in total ingredient weight and price reduction. This means that the proposed EA-PH-RWS-Avg is appropriate for shrimp diet formulation problem. Subsequently, in the final chapter, we conclude the whole research work on the shrimp diet formulation problem.

# CHAPTER 7

# CONCLUSIONS

The main aim of a diet formulation problem is to serve quality foods that meet the taste and fulfil the nutritional requirements with minimal cost. In this thesis, a diet formulation problem for animal was investigated and juvenile Whiteleg shrimp became the subject of the research. Evolutionary Algorithm (EA) was employed as a method to meet the objectives of this research subject to constraints that had to be considered.

The primary objective of this thesis is to develop a model that can lead to the creation of shrimp feed mix that will be able to meet the nutritional requirements for effective production. In order to achieve the main objective, some specific objectives need to be fulfilled. They are: (i) to identify the maximum and minimum requirements of shrimp feed for various aspects, (ii) to construct a new filtering heuristics known as Power Heuristics as part of the initialization procedure that is capable of filtering some combinations of ingredients from a selected database of choices, which could lead to a potentially poor solution, (iii) to construct a new crossover operator known as Average Crossover that is able to produce potentially good solution, and (iv) to conduct a comparative evaluation on the solutions based on several evolutionary models generated and what-if analyses.

## 7.1 Summary of diet formulation with Evolutionary Algorithm

Evolutionary Algorithms for diet formulation in aquaculture farming problem was first investigated by Furuya et al. (1997) and then Şahman et al. (2009). In this research, we continue their work and experiment with a more complex problem by considering more constraints. The shrimp farming industry is one of the vital industries that contribute to the economic growth since Malaysia is the ninth world producer producing 67505 tonnes of farmed shrimp in 2011. Therefore, this industry has to find a good strategy to reduce feed cost without compromising the shrimp's health. Formulation of the ingredients for shrimp diet is one of the important processes that help to reduce cost and at the same time satisfy nutritional needs.

In the formulation of the feed ingredients, we experimented with various EAs to produce a well-balanced diet for the shrimps based on a set of standard. We incorporated and tested several evolutionary operators in the basic EA, specifically three different selection operators and two different crossover operators. Initialization with and without the incorporation of Power Heuristics and Power Mutation operator was also experimented. Each operator at each stage of the EA was tested alternately while the other operators involved were controlled.

A heuristics procedure known as Power Heuristics was incorporated at the initialization stage in this EA model to explore the neighbourhood area when the initial solution was infeasible. The function of this Power Heuristics is to filter on some ingredients and remove it from calculation. This operator is capable of filtering some combinations of ingredients from a selected database of choices, which could lead to potentially poor solution. This heuristics works well if many options exist in the search space, which is suitable for a big problem like a shrimp diet formulation.

The testing of each of the three selection operators in this EA was done to evaluate and compare which selection operator is the best to be incorporated in the algorithm. Roulette Wheel Selection (RWS), Queen-Bee (QB) Selection, and Roulette-Tournament (RT) Selection were compared and the results indicated that the RWS is best suited to be included in the EA.

Subsequently, two crossover operators were experimented at the crossover stage of the EA. They were One-Point Crossover and Average Crossover. In comparison to the One-Point Crossover, the Average Crossover gave a good solution to the EA in terms of average penalty value, standard deviation, and the number of infeasible solutions obtained.

The Power Heuristics was again used together with Power Mutation at the mutation stage. In addition, elitism procedure was incorporated to keep the best found solution for each generation. The combination of all good operators was emphasized in the proposed Evolutionary Algorithm which collectively produces effective and efficient solutions in reasonable run times. Figure 7.1 shows the process flow of our proposed Evolutionary Algorithm.

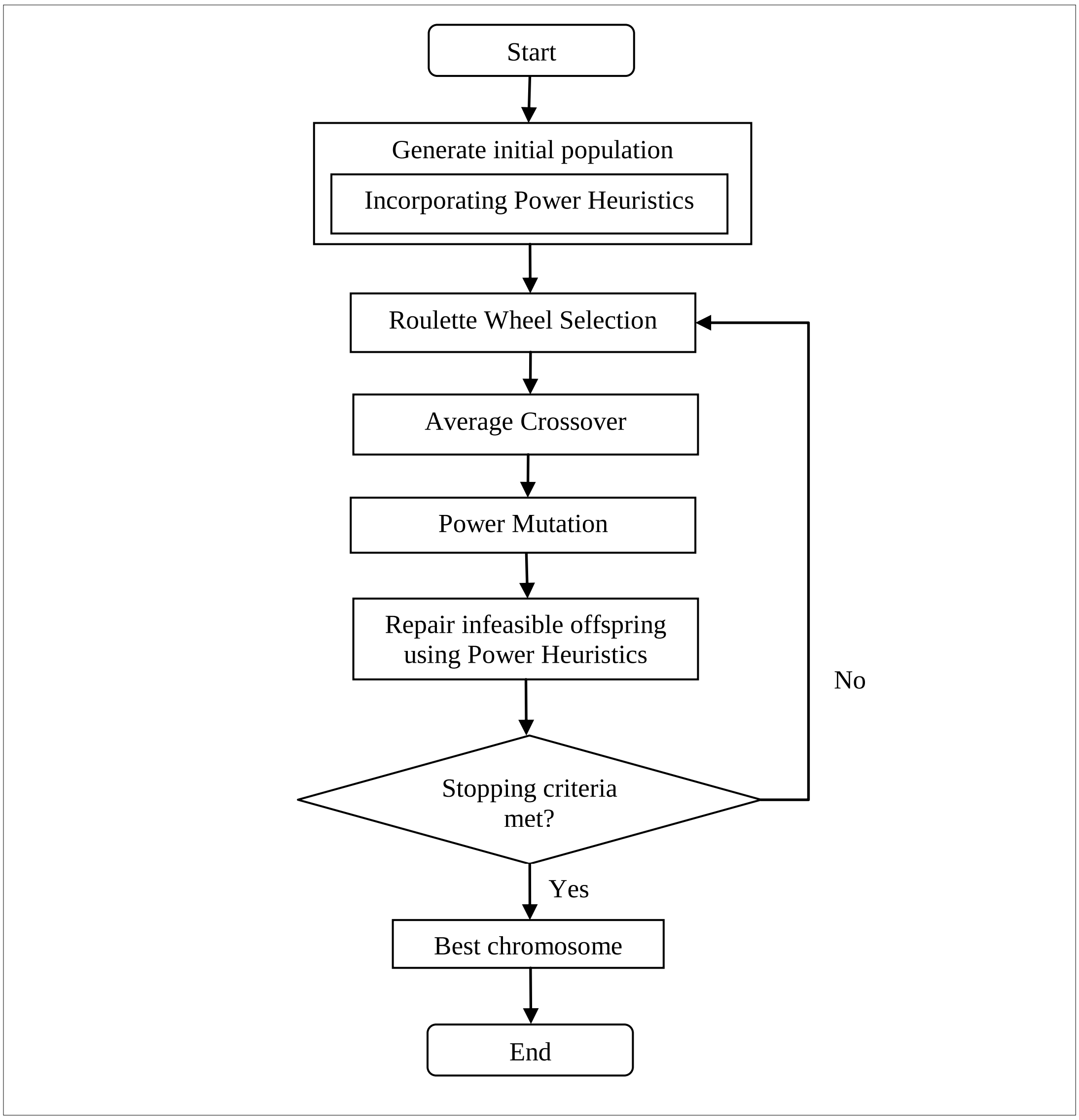



*Figure 7.1.* Proposed of Evolutionary Algorithm for Shrimp Diet Formulation

## 7.2 Accomplishment of research objectives

This research has fruitfully achieved all the identified research objectives, as described in the initial chapter. The primary objective of this research is to develop a model that can lead to the creation of shrimp diet formulation that is able to meet the nutritional requirements for effective production. Therefore, in order to reach the goal, we have identified four specific objectives. The first specific objective is to identify the maximum and minimum requirements of shrimp feed for various aspects. This was accomplished through data collection from literature review and interviews with experts, as described in Section 4.2.

The second specific objective is to construct a new filtering heuristics known as Power Heuristics as part of the initialization procedure that is capable of filtering some combinations of ingredients from a selected database of choices, which could lead to a potentially poor solution. This was accomplished in Section 5.2.2.2 and the results of its implementation are presented in Section 6.2.

The third specific objective is to construct a new crossover operator known as Average Crossover that is able to produce a potentially good solution. This objective was achieved via some modifications on the One-Point Crossover as described in Section 5.2.4.2. The results of its implementation were achieved in Section 6.4.2. The final specific objective is to conduct a comparative evaluation on the solutions based on several evolutionary models generated and what-if analyses were accomplished in Section 6.5 and 6.6.

This research has shown that our proposed Evolutionary Algorithm is effective and efficient for shrimp diet formulation problem. This EA approach leads to feasible high quality production of shrimp feed which is economically effective, and meets the nutritional requirements.

### 7.3 Contribution of the research

Our research has contributed modestly towards understanding the diet formulation problem. The discussion on contribution of this research is divided into three aspects: (i) contribution to the body of knowledge, (ii) contribution to practitioners, and (iii) benefits to policy makers. Contribution to the body of knowledge focuses on the concept of the methodology (i.e. evolutionary algorithm), while contribution to practitioners looks into the application of diet formulation problem and who will benefit directly in the aquaculture industry. Finally, this research benefits policy makers by showing the prototype achieved can help the Department of Fisheries Malaysia.

### 7.3.1 Contribution to the body of knowledge

In this research, we are able to propose three new evolutionary operators. The first one is the construction of a filtering heuristics known as Power Heuristics. This filtering heuristics is able to filter some combinations of ingredients from a selected database of choices, which could lead to a potentially poor solution. The Power Heuristics contributes to a new body of knowledge by providing an alternative procedure in enhancing a solution at the initialization stage.

Secondly, a new crossover operator known as Average Crossover is proposed in this study. A newly proposed Average Crossover that is able to produce a good solution adds to the class of crossover as suggested by other researchers, thus enriching the procedure of crossover in the literature. These two new contributions enrich the metaheuristics literature related to Evolutionary Algorithm, specifically and Artificial Intelligence, in general.

Thirdly, as a side contribution, we are able to construct a hybrid concept of selection known as Roulette-Tournament. This operator combines two existing selection techniques, which are the Roulette Wheel Selection and Binary Tournament Selection. Even thought the results is not as good as other establish selection operator, but our newly Roulette-Tournament Selection adds to the class of selection as recommended by other researchers, thus enriching the procedure of selection in the literature.

Lastly, we are able to propose a holistic enhanced model of the Evolutionary Algorithm with some enhancements in the evolutionary operators. This proposed model adds to another alternative of Evolutionary Algorithm, thus enriching the operation research and computational intelligence field. This proposed model can be used as an alternative to provide solution.

### 7.3.2 Contribution to practitioners

In terms of practice, this research has several advantages to practitioners and who are directly involved with shrimp diet. Firstly, this research is the first of its kind for solving a diet formulation problem in the Malaysian context by introducing a new approach to find a high quality contribution of ingredients for shrimp diet as a solution.

Secondly, instead of having only ingredients and nutrient restrictions, this research also includes a new criterion that is the bulk size of total ingredients weight based on user preference. It is designed to avoid wastage as our consideration of bulk size is not rigidly set at 100 kg. Thus, the user has the opportunity to design the appropriate budget depending on his/her needs. This is better than previous studies where the total ingredient weight is fixed at 100 kg or the users have to decide on the proportion so that the total weight is equivalent to 100%.

Thirdly, currently in Malaysia, feed or diet formulation technique applies Trial and Error method of combining ingredients via excel spreadsheet. Ten ingredients and six nutrients are taken into consideration in each diet formulation. Therefore, the formulation process of searching for the best diet combination is time consuming and not necessarily the best one. The result of our research can be used by nutritionists, farmers and manufacturers to formulate the diet for shrimp in a specific quantity. A database of ninety-one ingredients with sixteen nutrients including ten amino acids is considered in the diet formulation. This approach is able to produce good acceptable solutions in a short time. Practitioners can also test on many combinations of ingredients to get the best combination of nutritional needs with a reasonable cost quickly. Finally, the prototype developed allows anyone with limited knowledge in aquaculture nutrition to formulate the aquaculture diet.

### 7.3.3 Benefit to policy makers

Shrimps require many suitable ingredients with balanced nutrients for their growth. At present, the Department of Fisheries Malaysia only take into account six nutrients without considering amino acid in the formulation of shrimp diet. However, it is known that amino acid is part of protein that is very important for the shrimp health. Thus, by utilizing this research output, the department can consider more important nutrients for the benefit of the shrimp farming industry. In fact, many more combinations of ingredients can be considered with new local ingredients can be suggested. The best and suitable combination of appropriate ingredients will produce nutritious and palatable feed for shrimps.

### 7.4 Research limitations

Despite the best effort given to conduct this research, we acknowledge that this research has still a number of limitations. Nutrient values in each ingredient change over time due to several factors such as temperature and environment. We would like to consider these factors but due to the difficulties in obtaining relevant data from relevant sources, we have excluded these factors in our research. We also like to improve the user interface so that it can be more user friendly to facilitate ease of use. However, because the main goal of this research is to develop algorithm, the interface has been given less importance.

In this research, Power Heuristics function is to filter on some ingredients from fourteen selected ingredients out from the computation. It would be good if Power Heuristics can consider all the ingredients in the database without being restricted to any number and then offer a suggestion on the list of the ingredients. However, due to computational facilities, we restricted the number of ingredients to fourteen, which is based on literature and expert opinion.

### 7.5 Future work

In future research, variability on nutrients and price factors can be included to make the solution more significant. Our proposed Evolutionary Algorithm is able to combine with Monte Carlo Simulation to overcome these variation problems in a shrimp diet.  Model comparison on actual price fluctuates in different bulk quantities could be further investigated.

Shrimp diet as well as aquaculture diet is full of fuzzy elements since the actual feed eaten by them cannot be traced. This is the reason why the fuzzy concept can be included in the diet model in future research. This element perhaps is able to decrease the feed cost because the actual total ingredient weight produced might be reduced.

Fish meal is one of the most important feed since the taste is palatable and it is rich in nutrients especially protein. Thus, the more palatable feed might be obtained by including fish meal as a compulsory item in the formulation. In future research, this method can be used and the solution can be compared in terms of price and nutritive value. Besides, more nutrients can also be considered in future research based on the nutrient's priority as explained in Section 4.2.1. The inclusion of fatty acid and omega will produce a better combination of nutrients for shrimp requirement.

In improving the EA model, a selection concept based on mean and standard deviation can be investigated. Another potential work is by using fuzzy concept at the selection stage. All of these concepts will hopefully further improve the formulation of a diet combination. In fact, some adjustments in the mutation procedure can be improved to allow changes for the ingredients that are available in the database. This ability has the potential to ease the process of finding the best combination of the ingredients from all the available ingredients.

The newly proposed EA-PH-RWS-Avg Model can be tested using some relaxation of nutrient requirement to produce improved solution with less penalty value. Furthermore, a comparison with a data from previous researcher could be experimented to investigate the performance and effectiveness of the proposed model.

Finally, a prototype of the system can be generalized to other similar blending problem such as other aquaculture or livestock diet, baby milk and plant fertilizer. This can be done by making changes on nutrients' and ingredients' constraint. Moreover, interface of the prototype can be further improved to assist the user in order to use the prototype. The improved system might be used by feed processing companies, farmers and nutritionists to test the new proposed ingredients.

# REFERENCES


Abu Hassan, I., Hanafi, H.H., Che Musa, C.U., Pathmasothy, S. (1988, 25-29 October). Status of shrimp and finfish feeds in Malaysia. *Report of the Workshop on Shrimp and Finfish Feed Development*, Retrieved on April 5, 2010, from Food and Agriculture Organization (FAO) Corporate Document Repository.

Anderson, A. M., & Earle, M. D. (1983). Diet planning in the third world by linear and goal programming. *The Journal of the Operational Research Society, 34*, 9-16.

Afolayan, M. O., & Afolayan, M. (2008). Nigeria oriented poultry feed formulation software requirements. *Journal of Applied Sciences Research, 4*(11), 1596-1602.

Akiyama, D. M. (1992). *Future considerations for shrimp nutrition and the aquaculture feed industry.* Paper presented at the special session on shrimp farming, Louisiana, USA.

Al-Deseit, B. (2009). Least cost broiler ration formulation using linear programming technique. *Journal of Animal and Veterinary Advances, 8*(7), 1274-1278.

Alexander, D. L. J., Morel, P. C. H., & Wood, G. R. (2006). Feeding strategies for maximising gross margin in pig production. In J. D. Pintér, (Eds.), *Global optimization: scientific and engineering case studies* (pp. 33-43). New York: Springer.

Alkhazaleh, H. A. K. (2010). *Menu planning prototype system for Malaysian athletes*. Unpublished master's thesis, Universiti Utara Malaysia.

All about circuit. (2012). Retrieved on December 1, 2013 from http://www.allaboutcircuits.com/vol_5/chpt_4/11.html


Anderson, A. M., & Earle, M. D. (1983). Diet planning in the third world by linear and goal programming. *The Journal of the Operational Research Society, 34*, 9-16.


Aytug, H., & Koehler, G. J. (2000). New stopping criterion for genetic algorithms. *European Journal of Operational Research, 126* (3), 662-674.

Azeem, M. F., & Saad, A. M. (2004). Modified queen bee evolution based genetic algorithm for tuning of scaling factors of fuzzy knowledge base controller. Proceeding of The India Annual Conference (IEEE Indicon 2004) (pp. 299-303). Kharagpur, India.

Babić, Z., & Perić, T. (2011). Optimization of livestock feed blend by use of goal programming. *International Journal of Production Economics, 130*, 218-223.

Babu, G. P. & Murty, M. N. (1993). A near-optimal initial seed value selection in *k*-means means algorithm using a genetic algorithm. *Pattern Recognition Letters, 14*(10), 763-769.

Babu, S., & Sanyal, P. (2009). *Food security, poverty and nutrition policy analysis: Statistical methods and applications*. Washington, DC, USA: Academic Press.

Bäck, T. (2000). Binary strings. In T. Bäck, D. B. Fogel, & Z. Michalewicz (Eds.), *Evolutionary computation 1: Basic algorithms and operators* (pp. 132-135). Bristol: Institute of Physics Publishing.

Bailleul, P. J. D., Rivest, J., Dubeau, F., & Pomar, C. (2001). Reducing nitrogen excretion in pigs by modifying the traditional least-cost formulation algorithm. *Livestock Production Science, 72*, 199-211.

Balintfy, J. L., Ross, Z. T., Sinha, P.,& Zoltners, A. A. (1978). A mathematical programming system for preference and compatibility maximized menu planning and scheduling. *Mathematical Programming, 15*, 63-76.

Barbieri, M. A., & Cuzon, G. (1980). Improved nutrient specification for linear programming of penaeid rations. *Aquaculture, 19*, 313-323.

Bauer, R. J. (1994). *Genetic algorithms and investment strategies*. New York: John Wiley & Sons.

Bauer, W., Prentice-Hernandez, C., Tesser, M. B., Wasielesky Jr., W., Poersch, L. H. S.. (2012). Substitution of fishmeal with microbial floc meal and soy protein concentrate in diets for the pacific white shrimp *Litopenaeus vannamei*. *Aquaculture, 342–343, 112-116.*

Beasley, D. (2000). Possible applications of evolutionary computation. In T. Bäck, D. B. Fogel, & Z. Michalewicz (Eds.), *Evolutionary computation 1: Basic algorithms and operators* (pp. 4-19). Bristol: Institute of Physics Publishing.

Becker, G. S. (2008). *Livestock feed costs: Concerns and options.* The Library of Congress. Retrieved on March 27, 2012, from http://www.nationalaglawcenter.org/assets/crs/RS22908.pdf

Bhanja, U., Mahapatra, S., & Roy, R. (2013). QoT aware evolutionary programming algorithm for transparent optical networks.*International Journal for Light and Electron Optics*, *124*(23), 6391-6399.

Blickle, T. (2000). Tournament selection. In T. Bäck, D. B. Fogel, & Z. Michalewicz (Eds.), *Evolutionary computation 1: Basic algorithms and operators* (pp. 181-186). Bristol: Institute of Physics Publishing.

Blue, N., Brindle, B., Chase, D., Jaipaul, R., Machielse, M., Melvill, L., et al. (October 25, 2007). Economics and marketing: Marketing alternatives for feeder cattle. *Alberta Feedlot Management Guide: Alberta Agriculture Market Specialists.* Retrieved on April 18, 2011, from http://www1.agric.gov.ab.ca/$department/deptdocs.nsf/all/beef11849

Booker, L. B., Fogel, D. B. Whitley, D, Angeline, P. J.,& Eiben, A. E. (2000). Binary strings. In T. Bäck, D. B. Fogel, & Z. Michalewicz (Eds.), *Evolutionary computation 1: Basic algorithms and operators* (pp. 256-307). Bristol: Institute of Physics Publishing.

Burke, E. K., & Kendall, G. (2005). Introduction. In E. K. Burke, & G. Kendall (Eds.), *Search methodologies: Introductory tutorials in optimization and decision support techniques* (pp. 5-18). New York: Springer.

Cadenas, J. M., Pelta, D. A., Pelta, H. R., & Verdegay, J. L. (2004). Application of fuzzy optimization to diet problems in Argentinean farms. *European Journal of Operational Research*, *158*, 218-228.

Castrodeza, C., Lara, P., & Peña, T. (2005). Multicriteria fractional model for feed formulation: Economic, nutritional and environmental criteria. *Agricultural Systems*, *86*, 76-96.

Chakeredza, S., Akinnifesi, F. K., Ajayi, O. C., Sileshi, G., Mngomba, S., & Gondwe, F. M. T. (2008). A simple method of formulating least-cost diets for smallholder dairy production in sub-Saharan Africa. *African Journal of Biotechnology, 7*(16), 2925-2933.

Chang, P-C., Chen, M-H, Tiwari, M. K. & Iquebal, A. S. (2013). A block-based evolutionary algorithm for flow-shop scheduling problem. *Applied Soft Computing*, *13*(12), 4536-4547.

Chappell, A. E. (1974). Linear programming cuts costs in production of animal feeds. *Operational Research Quarterly (1970-1977), 25*(1), 19-26.

Chen, J. T. (1973). Quadratic programming for least-cost feed formulations under probabilistic protein constraints. *American Journal of Agricultural Economics, 55*(2), 175-183.

Chinneck, J. W. (2006). Chapter 14: heuristics for discrete search: Genetic algorithms and simulated annealing. In J. W. Chinneck (Ed.), *Practical optimization: A gentle introduction*. Ontario: Carleton University. Retrieved on January 12, 2013 from http://www.sce.carleton.ca/faculty/chinneck/po/Chapter14.pdf

Chipperfield, A. (1997). Introduction to genetic algorithms. In A. M. S. Zalzala, & P. J. Fleming (Eds.), *Genetic Algorithms in engineering systems*. (pp. 1-41). London: The Institution of Electrical Engineers.

Coello, C. A. (2002). Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: A survey of the state of the art. *Computer Methods in Applied Mechanics and Engineering, 191*, 1245-1287.

Cruz-Suárez, L. E., Nieto-L López, M., Guajardo-Barbosa, C., Tapia-Salazar, M., Scholz, U., & Ricque-Marie, D. (2007). Replacement of fish meal with poultry by-product meal in practical diets for Litopenaeus vannamei, and digestibility of the tested ingredients and diets. *Aquaculture, 272*, 466-476.

Dantzig, G. B. (1990). The diet problem. *Interfaces*, *20* (4), 43-47.

Das, S., Abraham, A., & Konar, A. (2009). *Metaheuristic clustering.* Chennai, India:Springer.

Davis, L. D. (1991). *Handbook of genetic algorithms.*New York: Van Nostrand Reinhold.

Deb, K. (2000a). Introduction to representations. In T. Bäck, D. B. Fogel, & T. Michalewicz (Eds.),*Evolutionary Computation 1: Basic Algorithms and Operators* (pp. 127-131). Bristol: Institute of Physics Publishing.

Deb, K. (2000b). Introduction to selection. In T. Bäck, D. B. Fogel, & T. Michalewicz (Eds.), *Evolutionary Computation 1: Basic algorithms and operators* (pp. 166-171). Bristol: Institute of Physics Publishing.

De Jong, K. A., & Sarma, J. (1993). Generation gaps revisited. In L. D. Whitney (Ed.), *Foundations of genetic algorithms 2*. San Mateo: Morgan Kaufmann.

Deep, K.,& Mebrahtu, H. (2011). Combined mutation operators of genetic algorithm for the traveling salesman problem. *International Journal of Combinatorial Optimization Problems and Informatics*, *2*(3), 1-23.

Deep, K.,& Thakur, M. (2007). A new mutation operator for real coded genetic algorithms. *Applied Mathematics and Computation, 193*(1), 211-230.

Deep, K., Singh, K., Kansal, M. L., & Mohan, C. (2009). A real coded genetic algorithm for solving integer and mixed integer optimization problems. *Applied Mathematics and Computation, 212*, 505-518.

Demment, M. W., Youngy, M. M., & Sensenig, R. L. (2003). Providing micronutrients through food-based solutions: A key to human and national development. *The Journal of Nutrition*, *133*, 3879-3885.

Deng, J., Mai, K., Ai, Q., Zhang, W., Wang, X., Xu, W., & Liufu, Z. (2006). Effects of replacing fish meal with soy protein concentrate on feed intake and growth of juvenile Japanese flounder, *Paralichthys olivaceus. Aquaculture*, *258*, 503–513.

Dobos, R.C., Ashwood, A.M., Moore, K., & Youman, M. (2004). A decision tool to help in feed planning on dairy farms. *Environmental Modelling and Software,19*, 967-974.

Duan, H-B, Xu, C. F. & Xing, Z. H. (2010). A hybrid artificial bee colony optimization and quantum evolutionary algorithm for continuous optimization problems. *International Journal of Neural Systems, 20* (*1*), 39-50.

Eckstein, E. (1970). Is the "diet problem" identical to the "menu planning problem"? *Management Science, 16*(9), 527-528.

Emmert, J. L., & Baker, D. H. (1997). Use of the ideal protein concept for precision formulation of amino acid levels in broiler diets. *Journal of Applied Poultry Research, 6*, 462-470.

Engelbrecht, E. (2008). *Optimising animal diets at the Johannesburg zoo.* Unpublished bachelor’s thesis, University of Pretoria, Pretoria, South Africa.

Erban, E. R. Jr., & Pruder, G. D. (1991). A method of economic comparisons for aquaculture diet development. *Aquaculture, 99*, 127-142.

Eshelman, L. J. (2000). Genetic algorithms. In T. Bäck, D. B. Fogel, & Z. Michalewicz (Eds.), *Evolutionary Computation 1: Basic algorithms and operators* (pp. 64-74). Bristol: Institute of Physics Publishing.

Li, F-C.,& Jin, C-X. (2008). Study on fuzzy optimization methods based on principal operation and inequity degree. *Computers and Mathematics with Applications, 56*, 1545-1555.

Food and Agriculture Organization of the United Nations (FAO). (2013). *Fisheries and Aquaculture Department, 2013* [FishStat Plus]. Available from FAO Web site, http://www.fao.org/fishery/statistics/software/fishstat/en

Food and Agriculture Organization (FAO). (2010). *The state of world fisheries and aquaculture.* Rome: Food and Agriculture Organization of The United Nations.

Fogel, D. B. (1997). The advantages of evolutionary computation. Proceeding of Bio-computing and Emergent Computation (pp 1-11). Skövde, Sweden.

Fogel, D. B. (2000a). Introduction to evolutionary computation. In T. Bäck, D. B. Fogel, & Z. Michalewicz (EDs.), *Evolutionary Computation 1: Basic algorithms and operators* (pp. 1-3). Bristol and Philadelphia: Institute of Physics Publishing.

Fogel, D. B. (2000b). Principles of evolutionary processes. In T. Bäck, D. B. Fogel, & Z. Michalewicz (Eds.), *Evolutionary Computation 1: Basic algorithms and operators* (pp. 23-26). Bristol and Philadelphia: Institute of Physics Publishing.

Forsyth, D. M. (1995). Chapter 5: Computer programming of beef cattle diet. In T. W. Perry, & M. J. Cecava (Eds.), *Beef cattle feeding and nutrition* (2nd ed., pp. 68). California: Academic Press, Inc.

Fox, J. M., Davis, D. A., & Lawrence, A. L. (2007). Methionine-, sulfur-containing amino acids: Limiting nutrients in commercial formulations? *Global Aquaculture Advocate, 10* (2), 74-77.

Fox, J. M., Davis, D. A., Wilson, M., & Lawrence, A. L. (2006). Current status of amino acid requirement research with marine penaeid shrimp. *Avances en Nutrición Acuícola VIII. VIII Simposium Internacional de Nutrición Acuícola* (pp. 182-196). Nuevo León, México.

Fox, J. M., Lawrence, A. L., & Li-Chan, E. (1995). Dietary requirement for lysine by juvenile penaeus vannamei using intact and free amino acid sources. *Aquaculture, 131*, 279-290.

Fox, J.M., Treece, G.D., & Sanchez, D.R. (2001). *Shrimp nutrition and feed management.* In M. C. Haws, & C. E. Boyd (Eds.), Methods for improving shrimp farming in Central America (pp 65-90). Managua, Nicaragua:University of Central America Press.

Furuya, T., Satake, K., & Minami, Y. (1997). Evolutionary programming for mix design. *Computers and Electronics in Agriculture, 18*, 129-135.

Gillespie, J. R., & Flanders, F. B. (2009). *Modern livestock and poultry production* (8th ed.). Clifton Park, New York, USA: Cengage Learning.

Glen, J. J. (1980). A mathematical programming approach to beef feedlot optimization. *Management Science, 26*(5), 524-535.

Glen, J. J. (1986). A linear programming model for an intensive beef production enterprise. *The Journal of the Operational Research Society, 37*(5), 487-494.

Goldberg, D. E. (1989). *Genetic algorithms in search, optimization, and machine learning*. United States of America: Addison-Wesley Publishing Company, Inc.

Golub, M. (1996). an implementation of binary and floating point chromosome representation in genetic algorithm. Proceedings of 18th International Conference ITI (pp. 417-422). Pula, Croatia.

Grefenstette, J. (2000). Rank-based selection. In T. Bäck, D. B. Fogel, & Z. Michalewicz (Eds.), *Evolutionary computation 1: Basic algorithms and operators* (pp. 187-194). Bristol: Institute of Physics Publishing.

Guoshuai, L., Kan, Y., Ran, Z & Jiao, Z. (2012). Application of genetic algorithm and annealing genetic algorithm in short-term optimal operation and economical operation of three gorges cascade. *Procedia Engineering, 28*, 81-84.

Guevara, V. R. (2004). Use the nonlinear programming to optimize performance response to energy density in broiler feed formulation. *Poultry Science, 83,* 147-151.

Hansard Team Kenya National Assembly. (1993, April 1, 1993). Kenya National Assembly: Parliamentary Debates, *National Assembly Debates: Official Report: Hansard, 1,* 626.

Hasan, B. H. F.,& Saleh, M. S. M. (2011). Evaluating the effectiveness of the mutation operators on the behaviour of genetic algorithms applied to non-deterministic polynomial problems. *Informatica, 35,* 513-518.

Hecker, F. T., Hussein, W. B., Paquet-Durand, O., Hussein, M. A.,& Becker, T. (2013). A case study on using evolutionary algorithms to optimize bakery production planning. *Expert Systems with Applications*, *40* (17), 6837-6847.

Herrera, F., Lozano, M., & Sánchez, A. M. (2005). Hybrid crossover operators for real-coded genetic algorithms: an experimental study. *Soft Computing, 9*, 280-298.

Herrera, F.,& Lozano, M. (2000). Gradual distributed real-coded genetic algorithms. *IEEE Transactions on Evolutionary Computation, 4,*43–63.

Hertrampf, J. W., Biswal, P. K.,& Mishra, S. K. (2005, June). *AllAboutFeed*. Retrieved on December 1, 2013, from http://www.allaboutfeed.net/Process-Management/Management/2005/6/A-revolutionary-concept-for-shrimp-feeding-AAF010209W/

Herva´ s-Martı´nez, C., & Ortiz-Boyer, D. (2005). Analyzing the statistical features of CIXL2 crossover offspring. *Soft Computing, 9*, 270-279.

Htun, M. S., Thein, T. T., & Tin, P. (2005). *Linear programming approach to diet problem for black tiger shrimp in shrimp aquaculture.* Paper presented at the Information and Telecommunication Technologies, Yangon, Myanmar.

Hung, Y., Chen, C., Shih, C., & Hung, M. (2003). Using tabu search with ranking candidate list to solve production planning problems with setups. *Computers & Industrial Engineering, 45,* 615–634.

The International Federation for Animal Health (IFAH). (2013). Retrieved June 16, 2011 from http://www.ifahsec.org/animal-health/our-story/

Jiao, J., Zhang, Y., & Wang, Y. (2007). A heuristic genetic algorithm for product portfolio planning. *Computers & Operations Research, 34,* 1777-1799.

Jung, S. H. (2003). Queen-bee evolution for genetic algorithms. *Electronic Letters, 39*(6), 575-576.

Ju, Z. Y., Deng, D-F, Dominy, W. (2012). A defatted microalgae (*Haematococcus pluvialis*) meal as a protein ingredient to partially replace fishmeal in diets of Pacific white shrimp (*Litopenaeus vannamei*, Boone, 1931. *Aquaculture, 354–355*, 50–55.

Karaboga, D. (2005). *An idea based on honey bee swarm for numerical optimization.* Kayseri/Türkiye: Erciyes University, Engineering Faculty.

Karaboga, D., & Basturk, B. (2007). A powerful and efficient algorithm for numerical function optimization: Artificial bee colony (ABC) algorithm. *Journal of Global Optimization, 39*, 459-471.

Karaboga, D., & Basturk, B. (2008). On the performance of artificial bee colony (ABC) algorithm. *Applied Soft Computing, 8*, 687-697.

Karci, A. (2004). Imitation of bee reproduction as a crossover operator in genetic algorithms. In C. Zhang, H. W. Guesgen, & W. K. Yeap (Eds.), *PRICAI 2004, LNAI 3157* (pp. 1015-1016). Springer-Verlag Berlin Heidelberg.

Kinnear Jr, K. E. (2000). Derivative methods in genetic programming. In T. Bäck, D. B. Fogel, & Z. Michalewicz (Eds.), *Evolutionary Computation 1: Basic algorithms and operators* (pp. 103-113). Bristol and Philadelphia: Institute of Physics Publishing.

Kock, H. C. D., & Sinclair, M. (1987). Multi-mix feedstock problems on microcomputers multi-mix feedstock problems on microcomputers. *The Journal of the Operational Research Society, 38*(7), 585-590.

Kornbluth, J. S. H., & Steuer, R. E. (1981). Multiple objective linear fractional programming. *Management Science, 27*(9), 1024-1039.

Lara, P., & Romero, C. (1992). An interactive multigoal programming model for determining livestock rations: An application to dairy cows in Andalusia, Spain. *The Journal of the Operational Research Society, 43*(10), 945-953.

Lara, P. (1993). Multiple objective fractional programming and livestock ration formulation: A case study for dairy cow diets in Spain. *Agricultural Systems, 41*, 321-334.

Lara, P., & Romero, C. (1994). Relaxation of nutrient requirements on livestocks rations through interactive multigoal programming. *Agricultural Systems, 45*, 443-453.

Lazo, J. P., & Davis, D. A. (2000). Ingredient and feed evaluation. In R. R. Stickney (Ed.), *Encyclopedia of Aquaculture* (pp. 453-463). Texas: Wiley-Interscience Publication.

Longman Dictionary of Contemporary English: The Living Dictionary (New Edition). (2003). Pearson Education Limited: Essex.

López-Pujalte, C., Bote, V. P. G. and Anegón, F, d., M. (2002). A test of genetic algorithms in relevance feedback. *Information Processing and Management, 38*, 793–805

Lozano, M., Herrera, F., & Cano, R. J. (2008). Replacement strategies to preserve useful diversity in steady-state genetic algorithms. *Information Sciences, 178,* 4421–4433.

Lu, X., & Zhou, Y. (2008). A genetic algorithm based on multi-bee population evolutionary for numerical optimization. *Proceeding of the 7th World Congress on Intelligent Control and Automation* (pp. 1294). Chongqing: IEEE.

Mathison, G. W. (1998). Book reviews: Animal feed formulation: Economics and computer applications. *Animal Feed Science Technology, 70,* 169-173.

Mayr, E. (1988). *Toward a new philosophy of biology: Observations of an evolutionist*. Cambridge, MA: Belknap.

McCall, J. (2005). Genetic algorithms for modelling and optimisation. *Journal of Computational and Applied Mathematics, 184*, 205-222.

Michalewicz, Z. (1994). *Genetic algorithms+data structures=evolution programs* (2nd ed.). New York: Springer-Verlag Berlin Heidelberg.

Michalewicz, Z. (2000a). Hybrid methods. In T. Bäck, D. B. Fogel, & Z. Michalewicz (Eds.),*Evolutionary computation 1: Basic algorithms and operators* (pp. 124-126). Bristol and Philadelphia: Institute of Physics Publishing.

Michalewicz, Z. (2000b). Introduction to constraint-handling techniques. In T. Bäck, D. B. Fogel, & Z. Michalewicz(Eds.),*Evolutionary Computation 2: Advanced algorithms and operators* (pp. 38-40). Bristol and Philadelphia: Institute of Physics Publishing.

Michalewicz, Z. (2000c). Repair algorithms. In T. Bäck, D. B. Fogel, & Z. Michalewicz(Eds.),*Evolutionary Computation 2: Advanced algorithms and operators* (pp. 38-40). Bristol and Philadelphia: Institute of Physics Publishing.

Michalewicz, Z.,& Schoenauer, M. (1996). Evolutionary algorithms for constrained parameter optimization problems. *Evolutionary Computation, 4*(1), 1–32.

Ministry of Agriculture, Food and Rural Affairs. (2008). Retrieved on September 26, 2012, from http://www.omafra.gov.on.ca/english/livestock/dairy/facts/08-039.htm.

Mitani, K., & Nakayama, H. (1997). A multiobjective diet planning support system using the satisficing trade-off method. *Journal of Multi-Criteria Decision Analysis, 6*, 131-139.

Mitchell, M. (1996). *An introduction to genetic algorithms.* London: The MIT Press.

Mohd Noor, N., Abdul Rahman, S., & Sulaiman, N. S. (2007). Meal planning model for nursery. Unpublished technical report, Universiti Utara Malaysia.

Mohd Razali, S. N. A. (2011). *Menu planning system for Malaysian boarding school using self-adaptive hybrid genetic algorithms*. Unpublished doctoral thesis, Universiti Utara Malaysia.

Mohr, G. M. (1972). The bulk constraint and computer formulations of least-cost feed mixes. *Review of Marketing and Agricultural Economics, 40* (1), 15-28.

Moraes, L. E., Wilen, J. E., Robinson, P. H., & Fadel, J. G. (2012). A linear programming model to optimize diets in environmental policy scenario. *Journal of Dairy Science, 95*, 1267-1282.

Munford, A. G. (1989a). A microcomputer system for formulating animal diets which may involve liquid raw materials. *European Journal of Operational Research, 41*, 270-276.

Munford, A. G. (1989b). Evaluating marginal costs associated with ratio and other constraints in Linear Programmes. *Journal of the Operational Research Society, 40*(10), 933-935.

Munford, A. G. (1996). The use of iterative linear programming in practical applications of animal diet formulation. *Mathematics and Computers in Simulation, 42*, 255-261.

National Research Council (NRC). (1982). *Nutritional data for United States and Canadian feeds (Third Revision): Unites States-Canadian tables of feed composition.* Washington, D. C.: The National Academic Press.

National Research Council (NRC). (2011). *Nutrient requirements of fish and shrimp.* Washington, D. C.: The National Academic Press.

Neumann, C., Harris, D. M., & Rogers, L. M. (2002). Contribution of animal source foods in improving diet quality and function in children in the developing world. *Nutrition Research, 22*, 193-220.

Nijkamp, P., & Spronk, J. (1978). Interactive multiple goal programming. *Report for the 7th meeting of the EURO-working group "Aide a la decision multicritere".* Retrieved on Jun 2, 2012, from Erasmus University Rotterdam.

Nguyen, T. T., Bouvarel, I., Ponchant, P., & van der Werf, H. M. (2012). Using environmental constraints to formulate low-impact poultry feeds. *Journal of Cleaner Production, 28*, 215-224.

Nott, H., & Combs, J. F. (1967). Data processing feed ingredient composition data. *Feed Stuffs, 39 (41)*, 21-22.

O'Connor, J. D., Sniffen, C. J., Fox, D. G., & Miligan, R. A. (1989). Least cost dairy cattle ration formulation model based on the degradable protein system. *Journal of Dairy Science, 72*, 2733-2745.

Oishi, K., Kumagai, H., & Hirooka, H. (2011). Application of the modified feed formulation to optimize economic and environmental criteria in beef cattle fattening systems with food by-products. *Animal Feed Science and Technology, 165*, 38–50.

Oloronfemi, T. O. S. (2007). Linear programming approach to least-cost ration formulation for poults. *Information Technology Journal, 6, 294-299.*

Oujifard, A., Seyfabadi, J., Kenari, A. A., Rezaei, M. (2012). Growth and apparent digestibility of nutrients, fatty acids and amino acids in Pacific white shrimp, *Litopenaeus vannamei*, fed diets with rice protein concentrate as total and partial replacement of fish meal. *Aquaculture*, *342–343*, 56–61.

Pan, W-T. (2012). A new fruit fly optimization algorithm: Taking the financial distress model as an example. *Knowledge-Based System, 26,* 69-74.

Pascual, C., Zenteno, E., Cuzon, G., Suárez, J., Sánchez, A., & Gaxiola, G. (2004). Litopenaeus vannamei juveniles energetic balance and immunological response to dietary proteins. *Aquaculture, 239*, 375-395.

Pathumnakul, S., Ittiphalin, M., Piewthongngam, K., & Rujikietkumjorn, S. (2011). Should feed mills go beyond traditional least cost formulation? *Computers and Electronics in Agriculture, 75*, 243-249.

Peña, T., Lara, P., & Castrodeza, C. (2009). Multiobjective stochastic programming for feed formulation. *The Journal of the Operational Research Society, 60*(12), 1738-1748.

Pesti, G. M., & Miller, B. R. (1993). *Animal feed formulation: Economics and computer applications.* New York: Chapman & Hall.

Pesti, G. M., & Seila, A. F. (1999). The use of an electronic spreadsheet to solve linear and non-linear "stochastic" feed formulation problems. *Journal of Applied Poultry Research, 8*(1), 110-121.

Pierre, N. R. S., & Harvey, W. R. (1986). Incorporation of uncertainty in composition of feeds into least-cost ration models. *Journal of Dairy Science, 69*, 3051-3062.

Piyaratne, M. K. D. K., Dias, N. G. J & Attapattu, N. S. B. M. (2012). Linear model based software approach with ideal amino acid profiles for least-cost poultry ration formulation. *Information Technology Journal, 11*(7), 788-793.

Polimeno, F., Rehman, T., Neal, H., & Yates, C. M. (1999). Integrating the use of linear and dynamic programming methods for dairy cow diet formulation. *The Journal of the Operational Research Society, 50*(9), 931-942.

Pomar, C., Dubeau, F., Létourneau-Montminy, M. P., Boucher, C., & Julien, P.-O. (2007). Reducing phosphorus concentration in pig diets by adding an environmental objective to the traditional feed formulation algorithm. *Livestock Science, 111*, 16-27.

Poojari, C. A.,& Varghese, B., (2008). Genetic algorithm based technique for solving chance constrained problems, *European Journal of Operational Research, 185*, 1128-1154.

Prékopa, A. (2003). Probabilistic programming. In Ruszczynski, A. & Shapiro, A. *Handbooks in OR and MS* (Vol. 10, pp. 267-351). North Holand: Elsevier Science.

Qing-dao-er-ji, R.,& Wang, Y. (2012). A new hybrid genetic algorithm for job shop scheduling problem. *Computers & Operations Research, 39*, 2291–2299.

Rahman, S. A., & Bender, F. E. (1971). Linear programming approximation of least-cost feed mixes with probability restrictions. *American Journal of Agricultural Economics, 53*, 612-618.

Ramli, R. (2004). An evolutionary algorithm for the nurse scheduling problem with circadian rhythms. Unpublished doctoral thesis, Universiti Sains Malaysia, Malaysia.

Randly, H. L, & Sue, H. E. (1998). Practical genetic algorithms. USA: A Willey-Interscience Publication.

Rehman, T., & Romero, C. (1984). Multiple-criterion decision making techniques and their role in livestock ration formulation. *Agricultural Systems, 15*(1), 23-49.

Rehman, T., & Romero, C. (1987). Goal programming with penalty functions and livestock ration formulation. *Agricultural Systems, 23*, 117-132.

Render, B., Stair, J., R. M., & Hanna, M. E. (2006). *Quantitative analysis for management* (9th ed.). Upper Saddle River, New Jersey: Pearson Education, Inc.

Romero, C., & Rehman, T. (2003). Livestock ration formulation via goal programming with penalty functions. In Romero, C. & Rehman, T. *Multiple criteria analysis for agricultural decisions* (Vol. 11, pp. 149-161).

Rossi, A., & Boschi, E. (2009). A hybrid heuristic to solve the parallel machines job-shop scheduling problem. *Advances in Engineering Software, 40*, 118-127.

Roush, W. B., Stock, R. H., Cravener, T. L., & D'Alfonso, T. H. (1994). Using chance-constrained programming for animal feed formulation at Agway. *Interfaces. The Institute of Management Sciences, 24*, 53-58.

Roush, W. B., Cravener, T. L., & Zhang, F. (1996). Computer formulation observations and caveats. *Journal of Applied Poultry Science, 5*, 116-125.

Roush, W. B., & Cravener, T. L. (2002). Stochastic true digestible amino acid values. *Animal Feed Science and Technology, 102*, 225-239.

Roy, L. A., Davis, D. A., Saoud, I., P., & Henry, R. P. (2007). Effects of varying levels of aqueous potassium and magnesium on survival, growth, and respiration of the Pacific white shrimp, Litopenaeus vannamei, reared in low salinity waters. *Aquaculture, 262,* 461-469.

Şahman, M. A., Çunkaş, M., İnal, Ş., İnal, F., Coşkun, B., & Taşkiran, U. (2009). Cost optimization of feed mixes by genetic algorithms. *Advances in Engineering Software, 40*, 965-974.

Salcedo-Sanz, S. (2009). A survey of repair methods used as constraint handling techniques in evolutionary algorithm. *Computer Science Review, 3*, 175-192.

Saxena, P., & Chandra, M. (2011). Animal diet formulation models: A review (1950-2010). In D., Hemming. *CAB reviews: Perspectives in agriculture, veterinary science, nutrition and natural resources* (1-9). UK: CABI.

Saxena, P. (2011). *Animal diet formulation using nonlinear programming: an innovative approach. Germany*: Lambert Academic Publishing.

Shang, Y. C. (1986). Research on aquaculture economics: A review. *Aquacultural Engineering, 5*, 103-108.

Shang, Y. C., Leung, P. & Ling, B-H. (1998). Comparative economics of shrimp farming in Asia. *Aquaculture, 164, 183-200.*

Sharief, S., Eldho, T., & Rastogi, A. (2008). Elitist GA based evolutionary algorithm for groundwater contaminant remediation using pump and treat method. Proceedings of The *12th International Conference of International Association for Computer Methods and Advances in Geomechanics (IACMAG)* (pp 2505-2512). Goa, India.

Shiau, S-Y. (1998). Nutrient requirement of penaeid shrimps. *Aquaculture, 164,* 77-93.

Smith, A. E., & Coit, D. W. (2000). Penalty functions. In T. Bäck, D. B. Fogel, & Z. Michalewicz (Eds.), *Evolutionary Computation 2: Advanced algorithms and operators* (pp. 41-48). Bristol and Philadelphia: Institute of Physics Publishing.

Sirisatien, D., Wood, G. R., Dong, M., & Morel, P. C. H. (2009). Two aspects of optimal diet determination for pig production: Efficiency of solution and incorporation of cost variation. *Journal of Global Optimum, 43*, 249-261.

Sivanandam, S. N., & Deepa, S. N. (2008). *Introduction to genetic algorithms*. New York: Springer.

Sivaraj, R., & Ravichandran, T. (2011). A review of selection methods in genetic algorithm. *International Journal of Engineering Science and Technology, 3*, 3792-3797.

Spears, W. M. (2000). Recombination parameters. In T. Bäck, D. B. Fogel, & Z. Michalewicz (Eds.), *Evolutionary Computation 2: Advanced algorithms and operators* (pp. 38-40). Bristol and Philadelphia: Institute of Physics Publishing.

Spears, W. M. & De Jong, K. A. (1991). An analysis of multi-point crossover. In J. G. E. Rowlins (Eds.), *Foundations of genetic algorithms.* (pp. 301-315). San Francisco: Morgan-Koufmann.

Spronk, J. (1981). *Interactive multiple goal programming: Applications to financial planning.* Boston: Martinus Nijhoff Publishing.

Sung, S. K., II-Hwan, K., Mani, V., & Hyung, J. K. (2008). Real-coded genetic algorithm for machining condition optimization. *International Journal of Advanced Manufacturing and Technology, 38*, 884-895.

Swanson, L. W., & Woodruff, J. G. (1964). A sequential approach to the feed-mix problem a sequential approach to the feed-mix problem. *Operations Research, 12* (1), 89-109.

Tacon, A. G. J. (1990, November). Fish feed formulation and production, *Report for the Project Fisheries Development in Qinghai Province.* Retrieved on March 31, 2012, from Food and Organization (FAO) Corporate Document Repository.

Tacon, A. G. J., Cody, J. J., Conquest, L. D., Divakaran, S., Forster, I. P., & Decamp, O. E. (2002). Effect of culture system on the nutrition and growth performance of Pacific white shrimp Litopenaeus vannamei (Boone) fed different diets. *Aquaculture Nutrition, 8*, 121-137.

Talbi, E-G. (2009). *Metaheuristics: from design to implementation*. New Jersey: John Wiley & Sons.

Taylor III, B. W. (2007). *Introduction to management Science.* New Jersey: Pearson Education.

Teufel, N., Markemann, A., Kaufmann, B., Zárate, A. V., & Otte, J. (2006). *Livestock production systems in South Asia and the Greater Mekong Sub-Region:A quantitative description of livestock production in Bangladesh, Cambodia, India, Lao PDR, Nepal, Pakistan, Sri Lanka, Thailand, and Viet Nam. Rome*: Food and Agriculture Organization.

Thammaniwit, S. & Charnsethikul, P. (2013). Bender's decomposition method for a large two-stage linear programming model. *International Transaction Journal of Engineering, Management and Applied Sciences and Technologies, 4*(4), 253-268.

Thomson, E., & Nolan, J. (2001). UNEForm: A powerful feed formulation spreadsheet suitable for teaching or on-farm formulation. *Animal Feed Science and Technology, 91*, 233-240.

Tozer, P. R. (2000). Least-cost ration formulations for holstein dairy heifers by using linear and stochastic programming. *Journal of Dairy Science, 83*, 443-451.

Tozer, P. R., & Stokes, J. R. (2001). A multi-objective programming approach to feed ration balancing and nutrient management. *Agricultural Systems, 67*(3), 201-215.

Udo, I. U., Ndome, C. B.,& Asuquo, P. E. (2011a). Use of stochastic programming in least-cost feed formulation for African catfish (*clarias gariepinus*) in semi-intensive culture system in Nigeria. *Journal of Fisheries and Aquatic Science, 6, 447-455.*

Udo, I. U., Ndome, C. B., Ekanem, S. B. & Asuquo, P. E. (2011b). Application of linear programming technique in least-cost ration formulation for African Catfish (*Clarias gariepinus*) in semi-intensive culture system in Nigeria. *Journal of Fisheries and Aquatic Science, 6, 429-437.*

van de Panne, C. V. D., & Popp, W. (1963). Minimum-cost cattle feed under probabilistic protein constraints. *Management Science, 9*(3), 405-430.

Venero, J. A., Davis, D. A., & Rouse, D. B. (2007). Variable feed allowance with constant protein input for the pacific white shrimp Litopenaeus vannamei reared under semi-intensive conditions in tanks and ponds. *Aquaculture, 269*, 490-503.

Villamide, M. J., Carobaño, R., Hernández, I., Rubio, J. M., Solis, I., Saiz, A. & Garcia, J. (2009). Evolution of a feed formulation practice in amandatory course on animal production. *Procedia Social and Behavioral Sciences, 1*, 1797-1801.

Wagner, J.,& Stanton, T. L. (2006). *Formulationg rations with the Pearson square*. Livestock series management. (Colorado University). Retrieved on December 1, 2013 from http://www.ext.colostate.edu/pubs/livestk/01618.pdf

Waugh, F. V. (1951). The minimum-cost dairy feed. *Journal of Farm Economics, 33*, 299-310.

Wilke, P., Gröbner, M., & Oster, N. (2002). A hybrid genetic algorithm for school timetabling. In B. McKay, & J. K. Slaney. *AI 2002: Advances in Artificial Intelligence* (Vol. 2557, pp. 455-464). New York: Springer Berlin / Heidelberg.

Xie, S-W., Tian, L. X., Jin, Y., Yang, H-J., Liang, G. Y., Liu, Y-J. (2014). Effect of glycine supplementation on growth performance, body composition and salinity stress of juvenile Pacific white shrimp, *Litopenaeus vannamei* fed low fishmeal diet. *Aquaculture, 418–419*, 159–164

Xu, C., Zhang, Q., Li, J., & Zhao, X. (2008). A bee swarm genetic algorithm for the optimization of DNA encoding. *The Proceedings of The 3rd International Conference on Innovative Computing Information and Control (ICICIC '08) (35).* Dalian, China.

Yang, X.-S & Deb, S. (2009). Cuckoo search via Levy flights. *The Proceedings of World Congress on Nature & Biologically Inspired Computing (NaBIC 2009)* (pp 210-214), India.

Yigit, T. (2007). Constraint-based school timetabling using hybrid genetic algorithm. In R. Basili, & M. T. Pazienza, *AI*IA 2007: Artificial intelligence and human-oriented computing* (Vol. 4733, pp. 848-855). Springer Berlin / Heidelberg.

Yu, H., Fang, H., Yao, P. & Yuan, Y. (2000). A combined genetic algorithm: simulated annealing algorithm for large scale system energy integration. *Computers and Chemical Engineering, 24,* 2023–2035.

Žgajnar, J., Juvančič, L., & Kavčič, S. (2008). *Spreadsheet tool for least-cost and nutrition balanced beef ration formulation.* Paper presented at the 16th Int. Symp. Animal Science Days, Strunjan, Slovenia.

Žgajnar, J., Juvančič, L., & Kavčič, S. (2009a). Combination of linear and weighted goal programming with penalty function in optimisation of a daily dairy cow ration. *Agricultural Economics, 55*(10), 492-500.

Žgajnar, J., Juvančič, L., & Kavčič, S. (2009b). Multi-goal pig ration formulation; Mathematical optimization approach. *Agronomy Research, 7*(2), 775-782.

Zhang, F., & Roush, W. B. (2002). Multiple-objective (goal) programming model for feed formulation: An example for reducing nutrient variation. *Poultry Science, 81*, 182-192.

Zhang, F. (1999). Stochastic models and software design for feed formulation. Unpublished doctoral thesis, The Pennsylvania State University.

Zhang, W., Xu, W.,& Gen, M. (2013). Multi-objective evolutionary algorithm with strong convergence of multi-area for assembly line balancing problem with worker capability. *Procedia Computer Science, 20, 83-89.*

Zhao, N., Wu, Z., Zhao, Y.,& Quan, T. (2010). Ant colony optimization algorithm with mutation mechanism and its applications. *Expert Systems with Applications, 37*, 4805–4810.

Zioganas, C. (1981). *The determination of viable, parity and optimum sizes of family-type sheep farms in the Epirus Region of Greece.* Unpublished doctoral thesis, Wye College-University of London.