ENHANCEMENT OF GENETIC ALGORITHM FOR DIABETIC PATIENT DIET PLANNING

HENG HUI XIAN

UNIVERSITI TEKNOLOGI MALAYSIA

ENHANCEMENT OF GENETIC ALGORITHM FOR DIABETIC PATIENT DIET PLANNING

HENG HUI XIAN

A project report submitted in partial fulfilment of the
requirements for the award of the degree of
Master of Engineering (Electrical – Computer and Microelectronic System)

Faculty of Electrical Engineering
Universiti Teknologi Malaysia

JUNE 2015

Dedicated, in thankful appreciation for the support, encouragement and understandings to my beloved parents, brother and sister.

ACKNOWLEDGEMENT

During the preparation of this thesis, I met with many people, researchers and academicians who had provided me with lots of advice and guidance. In particular, I would like to express my deepest appreciation to my thesis supervisor, Dr. Rabia Bakhteri for continuous support, guidance, critics and motivations throughout the entire project.

I would like to express my gratitude to all my family members who have been supporting and motivating me throughout my academic years. This academic year was not easy to complete especially to handle both work and academic workload at the same time. Hence, I am grateful to all my family members who support me all the way.

My sincere appreciation also to all my colleagues and friends who have provided me the advantageous suggestions, cooperation and technical support. Unfortunately, it is not possible to list all of them in this limited space. However, their assistance is highly appreciated.

ABSTRACT

Genetic Algorithm (GA) is an artificial intelligence (AI) based methodology for solving optimization problems. GA are problem dependent especially GA parameters and optimal parameter values require long experiment time. This project proposes a progress-value concept (PRGA) for crossover and mutation rate implement in steady-state GA (SSGA) to avoid trial and error experiment perform for optimal crossover and mutation rate. PRGA concept is using fitness value and total number of genes performed crossover and mutation for each individual within a generation to determine next generation crossover and mutation rate. PRGA is compare throughout SSGA with different fix crossover and mutation probability. The developed system is compiled using open source GA library (GAlib) for C programming language. Experimental results with proposed concept performance shows better processing time with SSGA.

ABSTRAK

Algoritma genetik (GA) merupakan satu kepintaran buatan (AI) yang berasaskan metodologi untuk menyelesaikan masalah pengoptimuman. GA adalah teknik yang khusus berdasarkan masalah terutamanya parameter GA dan nilai-nilai parameter optimum yang memerlukan masa latihan yang panjang. Projek ini mencadangkan satu konsep *progress-value* (PRGA) untuk kadar pindah silang dan mutasi dilaksanakan dalam *steady-state* GA (SSGA) untuk mengelakkan kaedah *trail-and-error* dalam mencari kadar pindah silang dan mutase yang optimum. Konsep PRGA menggunakan nilai kecergasan dan jumlah bilangan generasi melakukan pindah silang dan mutasi bagi setiap individu dalam tempoh satu generasi untuk menentukan kadar pindah silang dan mutasi kepada generasi akan datang. PRGA adalah membandingkan dengan SSGA menggunakan keberkasanan antara kadar tetap dan kadar boleh ubah bagi pindah silang dan mutasi. Sistem yang dibangunkan adalah disusun menggunakan *open-source GA library* (GAlib) untuk bahasa pengaturcaraan C. Keputusan eksperimen dengan prestasi konsep dicadangkan menunjukkan masa pemprosesan yang lebih baik dengan SSGA.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATION

AGA	-	Adaptive Genetic Algorithm
GA	-	Genetic Algorithm
GUI	-	Graphical User Interface
PRGA	-	Progress-Value Genetic Algorithm
SGA	-	Simple Genetic Algorithm
SSGA	-	Steady-State Genetic Algorithm

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 Project Background

Genetic Algorithm (GA) are relatively robust over many different types of search spaces. However, for particular problem domains, GA performance can often be improved by tuning their parameters such as type of operator, probabilities of applying the genetic operators, population size, *et al*. Often these parameters need to be optimized to allow the GA to deliver good and robust solutions for a whole family of similar problems. Achieving these goals requires in general a very careful and time-consuming fine tuning of parameters.

Diabetes is a common life-long health condition. From National Diabetes Statistics reported in 2014, American population with diabetes are increasing from year 2010, 25.8 million Americans or 8.3% to year 2012, 29.1 million Americans or 9.3%. In year 2010, from the 25.8 million Americans the figure for diagnosed and undiagnosed are 18.8 million and 7.0 million respectively. However year 2012, diagnosed Americans show 21.0 million and 8.1 million are undiagnosed.

From the statistic figure discuss in Figure 1.1, obviously amount of diabetes are increasing from year to year in different country. Diabetes happens when blood

glucose, also called blood sugar, is too high. Blood glucose is the main type of sugar found in our blood and is our main source of energy. Glucose comes from the food eaten and also made in our body's cells to use for energy.

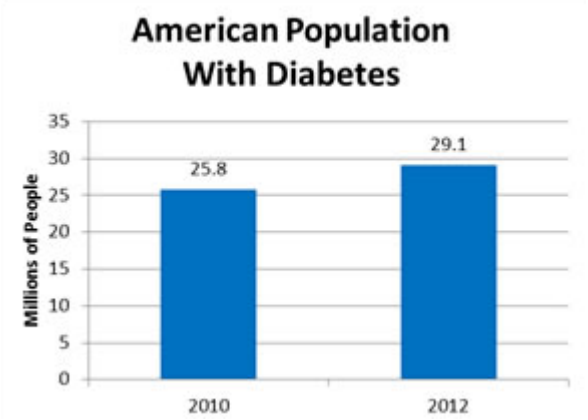

Figure 1.1: American Population with Diabetes reported in 2014

Pancreas is an organ helps with digestion and releases a hormone into our blood, called insulin. Insulin helps our blood carry glucose to all body's cell and sometimes if body does not produce enough insulin or insulin does not work as the way it should. Because of blood glucose levels increase hence glucose stays in our blood and does not reach cells, it cause diabetes or maybe pre-diabetes. Therefore diabetes patients need to plan and prepare their diet according to maximum glucose can be taken per meal or day.

Furthermore, diabetic patient can control their blood glucose level through exercise or meal and treatments. When taking meal, most of the patient will according to the amount or food item been selected by doctor. Other than that, patient also can follow the recipe provided through internet and calculate the amount can be taken according personally. Diet planning using genetic algorithm will discuss detail in the project so that patient can have the food item interest with the result amount show from system.

1.2 Problem Statement

Genetic Algorithm is an iterative process and the operational number of generations is not easily determined. However, genetic parameters are problem dependent and optimal parameter values require long experimentation time. The power of genetic algorithm arises primarily from crossover and mutation. As explain crossover operation is used to generate offspring by exchanging bits in a pair of individual called parents chosen from the population, with the probability the good solution can generate better ones. The mutation operator is used to change some genes in selected individual with a mutation probability or mutation rate which leading to additional genetic diversity to help the search process escape from local optimal traps. From problem to problem genetic parameters value for crossover and mutation rate dependent.

Normally patient have to calculate their total carbohydrate intake per meal by basic calculation according maximum carbohydrate intake recommend or follow the food recipe provided internet. Example, if patient wish to have a spaghetti and a cup of soft drink with apple fruits for he/she lunch meal, he/she need to consider and calculate the total amount of sugar for the food portion. However, patient will need to sacrifice foods or reduce food portion to achieve the maximum carbohydrate can be taken per meal. It is time consuming and need sacrificing food items to achieve carbohydrate amount.

Diabetics patients need a nutrition table side-by-side while meal planning and preparation to calculate total sugar included for each meal and items. Example, user need to consider that one cup of milk is equal to half cup of fruit juice shown in Figure 1.2, so as a diabetic patient they need to consider widely that are any other food can replace the current food have been chosen. This is not user friendly and inconvenience.

Figure 1.2: Food amount contain 15g of carbohydrate

1.3 Objectives

The main objective of this project is to enhance genetic algorithm with adaptive probability crossover and mutation for optimal number of generation. The second objective is to design a software-based genetic algorithm for implementation in a computational nutrition diet planning for diabetic patient based on food item and suggest user amount of food portion can be taken with total less than maximum target carbohydrate amount per meal.

The following goals are to be achieved during the whole progress of this project.

I. To reduce number of generations and execution time to obtain optimal food amount.
II. To ensure the objective value with match target value within tolerance 0.1%.
III. To prove simple linear equation can be solve by using Genetic Algorithm.
IV. To design and develop software of Genetic Algorithm using C++.

V. To integrate the diet planning software with a comprehensive open source food nutrition database (NutriDB) that content foods carbohydrate amount.

1.4 Scope of Project

Firstly is to understand the methodology and algorithm of Genetic Algorithm (GA) using hand calculation and MATLAB. Variables are carbohydrate value for each food item selected, fix constant or constraint is the maximum carbohydrate amount advice from doctor. Implement genetic algorithm using Visual C++ language.

Design genetic algorithm using console C++ with only 20 items of food. Design a user friendly interface using Window Visual C++ that maps food items from nutrition database (NutriDB). Use open source GA library (GAlib) and customize genetic algorithm to suit this project. Customize genetic algorithm is scalability to assign and tune useful genetic parameters especially crossover and mutation rate.

Integrate nutrition database (NurtiDB) with diet planning tool with 10k of food items. Understand from MySQL database to read nutrition information for each food. Understand how to extract data from database table and form a nutrition calculation formula and ready carbohydrate value for objective function to be used in genetic algorithm.

1.5 Thesis Organization

Chapter 1 provides project background of diabetic describe briefly and problem facing on diabetic patient been listed. The purpose of this chapter is to describe the objectives of this project to solve the problem for diabetic patient and genetic algorithm with scope of project. Chapter 2 include theory and background of Genetic Algorithm and shortly describes some related works of mathematical equation using Genetic Algorithm.

Identify and explanation of using Genetic Algorithm for the project are presented in Chapter 3. Furthermore, the adaptive genetic algorithm, genetic algorithm flow chart and the variables been selected in this project for C++ language are describes detail accordingly. Chapter 4 is based on result get from solving a generation of simple linear equation using hand calculation and discuss for each expected result. However, all results and discussion of each step output from command prompt window using C++ are described. Result and discussion using adaptive genetic algorithm also included. Chapter 5 conclude the overall project and the enhancement can be used in future work for this project.

REFERENCE

1. Hermawanto, D. (2013). Genetic Algorithm for Solving Simple Mathematical Equality Problem. *arXiv preprint arXiv:1308.4675*.
2. Patel, Roshni .V, and Jignesh. S Patel. 2011. 'Optimization Of Linear Equations Using Genetic Algorithms'. *IJAR* 2 (3): 56-58. doi:10.15373/2249555x/dec2012/19.
3. F.M.PATEL and N. B. PANCHAL. 2012. 'The Matlab Solution Of Mathematic Equation Using Genetic Algorithm For Optimum Result'. *Global Journal For Research Analysis* 3 (1): 45-47. doi:10.15373/22778160/january2014/68.
4. Nayak, T., & Dash, T. (2013). Solution to Quadratic Equation Using Genetic Algorithm. *arXiv preprint arXiv:1306.4622*.
5. Mhetre, Punam S. "Genetic algorithm for linear and nonlinear equation."*International Journal of Advanced Engineering Technology* 3, no. 2 (2012): 114-118.
6. Ikotun Abiodun, M., Lawal Olawale, N., & Adelokun Adebowale, P. The Effectiveness of Genetic Algorithm in Solving Simultaneous Equations.*International Journal of Computer Applications (0975–8887) Volume*.
7. Yi Wang. Using Genetic Algorithm for Parameter Estimation. 30th September 2004. China.
8. Yang, M. D., Yang, Y. F., Su, T. C., & Huang, K. S. (2014). An efficient fitness function in genetic algorithm classifier for landuse recognition on satellite images. *The Scientific World Journal*, *2014*.

9. A technique for constructing an initial point for solving a system of linear equations by iterative methods. In *Circuits and Systems, 1990., IEEE International Symposium on* (pp. 1589-1590). IEEE.
10. Srinivas, M., & Patnaik, L. M. (1994). Adaptive probabilities of crossover and mutation in genetic algorithms. *Systems, Man and Cybernetics, IEEE Transactions on, 24*(4), 656-667.
11. Hong, T. P., Wang, H. S., Lin, W. Y., & Lee, W. Y. (2002). Evolution of appropriate crossover and mutation operators in a genetic process. *Applied Intelligence, 16*(1), 7-17.
12. Dastkhosh, A. R., Dadashzadeh, G., & Sedaaghi, M. H. (2010). New Design Method of UWB Microstrip Filters Using Adaptive Genetic Algorithms with Defected Ground Structures. *International Journal of Microwave Science and Technology, 2010*.
13. Gonçalves, J. F., & Resende, M. G. (2012). A parallel multi-population biased random-key genetic algorithm for a container loading problem. Computers & Operations Research, 39(2), 179-190.
14. Haupt, R. L., & Haupt, S. E. (2004). Practical genetic algorithms. John Wiley & Sons.
15. Aggarwal, S., Garg, R., Dr. Goswami, P. (2014). A Review Paper on Different Encoding Schemes used in Genetic Algorithms. *International Journal of Advanced Research in Computer Science and Software Engineering*, 4(1), 596-600.
16. Wall, Matthew Bartschi. *A genetic algorithm for resource-constrained scheduling.* PhD diss.. Massachusetts Institute of Technology; 1996.