IMPROVING TOWER DEFENSE GAME AI
(GENETIC ALGORITHM VS
GENETIC PROGRAMMING)

TIO CHUN CHIENG

FACULTY OF COMPUTING AND INFORMATICS
UNIVERSITY MALAYSIA SABAH

2015

IMPROVING TOWER DEFENSE GAME AI
(GENETIC ALGORITHM VS
GENETIC PROGRAMMING)

TIO CHUN CHIENG

**THESIS SUBMITTED IN PARTIAL FULFULMENTS
OF THE REQUIREMENTS FOR THE DEGREE OF
BACHELOR OF COMPUTER SCIENCE
(SOFTWARE ENGINEERING)**

**FACULTY OF COMPUTING AND INFORMATICS
UNIVERSITY MALAYSIA SABAH**

2015

DECLARATION

I hereby declare that this thesis, submitted to Universiti Malaysia Sabah as partial fulfillment of the requirements for the degree of Bachelor of Computer Science (Software Engineering) has not been submitted to any other university for any degree. I also certify that the work described herein is entirely my own, except for quotations and summaries sources of which have been duly acknowledged.

This thesis may be made available within the university library and may be photocopied or loaned to other libraries for the purposes of consultation.

6 July 2015 Tio Chun Chieng

CERTIFIED BY

Dr Chin Kim On
SUPERVISOR

ACKNOWLEDGEMENT

I would like to appreciate my supervisor Dr. Chin Kim On for all the guidance he gave me. Besides that, I also want to thank my family and friend who always gives me support.

I would like to thank all my lecturers that taught me since my first year in UMS. They are the one who teach me all the precious knowledge that I need in this research.

ABSTRACT

The processes of designing and developing in digital game is costly. Tower defense games received much attention recently. However, there are few challenges found in designing the maps used in the game; the maps used is either too easy to be played or too difficult to be won. It happened as programmers simply developed the maps without proper planning and testing. This research proposed a technique using Evolutionary Algorithms (EAs) and Artificial Neural Networks (ANNs) to auto generate controllers to test the proposed maps. The proposed method can lead to significantly better intelligent system rather than depending on either EAs or ANNs alone. The selected EAs are Genetic Algorithm (GA) and Genetic Programming (GP) and the selected ANNs are Feed-Forward Neural Network (FFNN), Elman Recurrent Neural Network (ERNN), Jordan Recurrent Neural Network (JRNN), and an Ensemble Neural Network (ENN). The proposed ENN is a weighted sum NN composed of single FFNN, single ERNN, and single JRNN. The elitism concept is integrated in the optimization processes. The GA utilized uniform mutation and uniform crossover operators whereas the GP used one point mutation and one point crossover operators. Each experiment is conducted 10 times for each algorithm. The boxplot and t-test are used as performance metric in this project. As a result, by comparing the proposed EAs with same ANN used, the results showed GA performed slightly better than GP. On the other hand, by comparing different ANNs in the GA experiment, GA hybrid ENN achieved 91% of success rate, GA hybrid FFNN achieved 89% of success rate, GA hybrid JRNN achieved 81% of success rate, and GA hybrid ERNN achieved 70% of success rate. Hence, it concludes that GA hybrid ENN outperformed other type of NNs. Interestingly, the GP experiments showed different results. The GP hybrid FFNN achieved highest average success rate which is 88%, GP hybrid ENN achieved 67% success rate, GP hybrid ERNN achieved 64% success rate, and GP hybrid JRNN achieved 63% success rate. Overall comparison showed GA hybrid ENN outperformed the GP hybrid FFNN.

ABSTRAK

Proses merekabentuk dan pembangunan dalam permainan digital adalah mahal. Permainan pertahanan menara mendapat perhatian baru-baru ini. Walau bagaimanapun, terdapat beberapa cabaran yang terdapat dalam merekabentuk peta yang digunakan dalam permainan; peta yang digunakan terlalu mudah untuk dimainkan atau terlalu sukar untuk dimenangi. Ia adalah disebabkan oleh pembangunan peta tanpa perancangan dan ujian yang betul. Projek ini menggunakan algoritma evolusi dan rangkaian neural buatan untuk menjana alat kawalan secara automatik serta menguji peta yang dicadangkan. Gabungan antara rangkaian neural buatan dan algoritma evolusi boleh membawa kepada sistem yang lebih baik daripada bergantung kepada rangkaian neural buatan atau algoritma evolusi sahaja. Algoritma evolusi yang terpilih adalah Genetic Algorithm dan Genetic Programming manakala rangkaian neural buatan yang terpilih adalah rangkaian neural Feed-Forward, rangkaian neural Elman Recurrent, rangkaian neural Jordan Recurrent, dan rangkaian neural Ensemble. Pemilihan induk untuk kedua-dua algoritma evolusi adalah pemilihan elitisma. Mutasi dan kaedah silang yang seragam digunakan untuk Genetic Algorithm manakala mutasi dan kaedah silang satu titik digunakan untuk Genetic Programming. Setiap eksperimen telah dijalankan selama 10 kali untuk setiap algoritma. Boxplot dan ujian T telah digunakan sebagai penilaian metrik dalam projek ini. Dengan membandingkan algoritma evolusi yang berbeza dengan rangkaian neural buatan yang sama, Genetic Algorithm mencapai prestasi yang lebih baik daripada Genetic Programming. Dengan membandingkan Genetic Algorithm yang sama dengan rangkaian neural buatan yang berbeza, rangkaian neural Ensemble (91.00%) mencapai keputusan yang terbaik manakala rangkaian neural Elman Recurrent (70.00%) mencapai keputusan yang paling teruk. Dengan membandingkan Genetic Programming yang sama dengan rangkaian neural buatan yang berbeza, rangkaian neural Feed-Forward (88.00%) mencapai keputusan yang terbaik manakala rangkaian neural Jordan Recurrent (63.00%) mencapai keputusan yang paling teruk. Sebagai kesimpulan, Genetic Algorithm Ensemble Neural Network mencapai prestasi yang lebih baik daripada algoritma lain-lain dalam projek ini.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF GRAPH

LIST OF EQUATION

CHAPTER 1

INTRODUCTION

1.1 Overview

Chapter 1 introduces the general overview and background for this research. This chapter is divided into six sections. First section is a general overview describing the content of this chapter. The next section gives a brief explanation on the background of the project. Next, problem statement will be discussed in section three. Then, section four lists the main objectives of this research. The scope of the project will be discussed in section five and the project contribution will discussed in section six. The final section states the organization of other chapter for this project.

The main intention of this project is to create a new customized tower defense map with different pathway and implement different evolutionary artificial neural network model. In between of each different neural network models, results of each model will be collected and analyzed after few times of experiment. The experiment will be repeated until optimization of result is achieved. At the end of this project, project summary will be stated.

1.2 Project Background

There are many different types of tower defense game that are currently being played. But, the basic concept of tower defense is always the same because it is one genre of the RTS game. Generally, player has to build a tower in the building region to attack the creeps approaching to their base. The tower is built based on the player's money or gold and the money will increase when the tower successfully kills the creeps. The player needs to learn where to build the tower at a strategic place – the junction of the path, in order to maximize the number of creeps killed. The tower can be upgraded or a stronger tower can be built when the player has earned enough gold. The player has several types of towers available, each with various attributes and side-effects. The attack range of tower mostly will affect the result of game. This is because the tower is unable to shoot at anything when there are no creeps in their shooting range. On the other sides, different types of creeps have their own abilities and some will resist certain types of tower attack. If there are a specific number of creeps reaching their base, the player will lose the game.

Tower defense games are continuously being developed and change the way how games are being played. Due to this reason, player would usually find this type of game interesting and challenging. Unfortunately, in the single – player part the AI in these games does not learn these new strategies. This will lead to the player become superior to the computer AI after playing for a long period and adapt to the game strategies. This also make human player loses their interest in the single player part of the game. If the AI in these tower defense game can adapt itself to the new strategies and earn new strategies on its own against the player strategies, player will continue their passion in the game due to the challenge and varied game strategies.

1.3 Problem Statement

By referring back to senior's research, they have used Genetic Algorithm, Genetic Programming, Feed Forward Recurrent Neural Network and Elman Recurrent Neural Network in their research. The result obtained was categorized into a few parts.

- By comparing different EA,
    - GAFFNN performs better than GPFFNN whereas GAERNN performs better than GPERNN.
    - The fixed size of GP binary tree causes GP slower than GA to reach global optimum. This is explaining the performance of GP is less well than GA in their experiment.

- By comparing each model with different ANN,
  - GAFFNN performs better than GAERNN. This is because the search space of ERNN is larger than FFNN. It causes ERNN hard to find near-optimal and jump to local optima.
  - Next, the GPERNN is performs better than GPFFNN. The depth of binary tree of GP is fixed to suit with the total number of ANN's weight and this limits the global search of GP. However, search space of ERNN is wider than FFNN. So, GPERNN manages to find local optima faster if compare to GPFFNN.

After analyzing the experimental results, the combinational of evolutionary algorithm and artificial neural network affect the performance of a particular model.

Running time of experiment is also another problem for senior's research. In their project, four experiments taken up with each experiment has 10 runs. Each test will run 150 generations as termination condition of this experiment. A total 40 runs had been implemented. It requires nearly 24 hours for each run to be fully conducted and it requires approximately 40 days to run all experiments. Many other experiments setup cannot be tested due to insufficient amount of time.

There was less research paper available for the tower defense game. Although several of map design is available, but researcher seldom put so much effort on the level of difficultly in game design. The level of difficulty of game can beneficial to game programmer to reduce the time required in designing the game AI. This also help to improve the game play experience. Generally, level of difficulty divided into three layers: low/easy level, middle/intermediate level, and high/hard level. The AI units will be generated based on the players' selection. Level of difficulty can be determine after calculate success rate of game play through the experiment.

Table 1.1: Example of Level of Difficulty Based on Success Rate in Game Play

Success Rate (%)	Level of Difficulty
0 - 30	High/hard
30 - 70	Middle/intermediate
70 – 100	Low/easy

1.4 Project Objectives

The project objectivities are as follows:

- Design and customized a new tower defense maps from World Editor Warcraft 3 for optimization and testing used.
- To investigate, design, implement, and compare the selected evolutionary algorithms (genetic algorithms and genetic programming) to evolve for the required game controllers.
- To investigate, design and implement, and compare the selected artificial neural networks (Feed Forward Neural Network, Elman Recurrent Neural Network, Jordan Recurrent Neural Network, and Ensemble Neural Network) in evolving the required game controllers.
- To compare result of Feed Forward, Elman Recurrent, Jordan Recurrent, and Ensemble Neural Network.

1.5 Project Scope

The project scopes are as follow:

- The game controller chosen for project is the Warcraft 3 World Editor, a platform that allows us to implement the AI and environment. Hence, the usability is limited only in Warcraft 3 World Editor.
- There are 30 regions to build the towers and each game is limited to waves of creeps which contain 20 creeps per wave.

1.6 Project Contribution

The section shows the contribution of this project and the previous senior's work. The comparison is shown in the table below:

Table 1.2: Contribution Project Work

Contribution Project Work	
Senior' work	Currently work
- Using the algorithm GAFFNN, GAERNN, GPFFNN, and GPERNN.	- Using the algorithm GAJRNN, GAFFNN, GAERNN, GPJRNN, GPFFNN, and GPERNN.

- Conducted their experiment with using the existing map which created by them.	- Conducted the experiment by using new map which have a new design path and used as new test-bed.
- Not including the ensemble neural network	- Including the ensemble neural network.

1.7 Project Organization

Chapter one is the introduction which concisely introduces general overview of the project and comprises the reviewing content of the chapter, problem background, problem statement, objectives, project scopes and also organization of the project.

Chapter two is literature review which discusses evolutionary algorithms, artificial neural networks, and evolutionary – artificial neural network. This chapter also comprised the past researches which were done by the previous researches and description of some popular TD game.

Chapter three is methodology. It reviews and discusses method used and overall framework conduct in building this project. The project plan, i.e. selected evolutionary algorithms and artificial neural networks, are included at here. This chapter ends with the summary of the experimental plan.

Chapter four is the system design and implementation. The system is analysis and design. All algorithms and parameters are discussed at this chapter.

Chapter five is the experiment result and analysis. The result of experiment is collected and analysis the data by using the graphs, tables and boxplots to make the result more readable and clearly.

Chapter six is the conclusion which is also is the last part for this project. It will summarize the project and result obtained. A short summary chapter review in this chapter will also be included.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

This chapter will discuss previous research which had been done by other researchers related with the evolving and developing of artificial intelligence in tower defense game. Firstly, we will start RTS genre game – Tower Defense and provide some other examples of tower defense game. Then, we will further discuss about the research methods used in tower defense game since last decade. The motivation of research and summary of chapter will be concluded at the end of chapter.

2.2 Real Time Strategy Game

Most of the RTS game provided a great ideal platform for Artificial Intelligence (AI) research. There are some of researches provided some evidences about AI are getting more popular in the RTS games. For example, first person shooting game AI development by implement genetic algorithm which are conducted by Zanetti and Rhalibi (2004) and Cole, Louis, and Miles (2004). Louis and Miles (2005) developed a Case-Injected genetic algorithm for Non-Player Character action in RTS game. Hsieh and Sun (2008) trained the decision system in one singe battlefield by used 300 professional game players' replays. The game AI approach is something like "God mode" because of AI can predict all the possible user actions and strategy to play the game. In RTS game, there are three main strategies and counter strategies. First strategy is fast attack which meant player will try to take out opponent after only few minutes of playing. Next strategy is learning new technology quickly, where allow player to build units that are much stronger than player opponent units. Last strategy is based on speed earning money and getting economy up then player to build stronger units than opponents' later in game. For sure there will a lot of variations and combinations of strategies continuously being developed in future.

2.3 Tower Defense

Tower defense (TD) is one genre of the RTS game and relatively simple. The objective of the game is to prevent specifies number of enemies from reaching the base. That is only way the player can do is to build the tower along a fixed path of customized map to attack those enemies move forward to their base. There is have a few waves of creeps during each level of game and each wave is released when each enemy in the current wave has either been removed or has reached the base. The property of tower that affects the AI the most is range attack of tower. Therefore, tower will be useless when there is not contain any creeps in the attack range. The stronger tower will worth costly and more money can earn by killing each enemy.

Tower defense games provide a superior field to study AI for several reasons. Firstly, the game is simple enough to implement the AI to be tested on the entire game whereas the more complex of the game will caused require splitting components apart and studying separately. Tower defense game also easy to set up if compare with the other games. Many of tower defense game require tactical and strategic to be successful but usually this will provide a balanced learning curve. RTS games are a popular genre, which has been worked on for years, but RTS AI performance is horrible compared to humans (Paul, 2011). Although RTS games are far more complex, lessons learned on tower defense games can be extended to improve the quality of AI in games (Buro and Furtak, 2003).

Tower defense games had been growing in popularity with the use of flash web games. There are some noticeable tower defense games which are Bloons Tower Defense 5, Plant VS Zombie, Warzone, and etc.

2.3.1 Bloons Tower Defense 5

Bloons Tower Defense 5 is the most popular flash Tower Defense game. This game uses a few customized path on map design and categorize it into easy, medium, hard and etc. The interesting part of this game is that a new tower will be unblocked after the game has reached a particular level. The tower in this game has many different attribute like one kind of tower can slow the movement of enemy to move toward the base. Other towers fire multiple attacks, attacks in two directions, attacks that damage the area of the enemy. Special features of this game are such as towers that can locate hidden creeps and another tower that is able to send creeps back to the starting position.

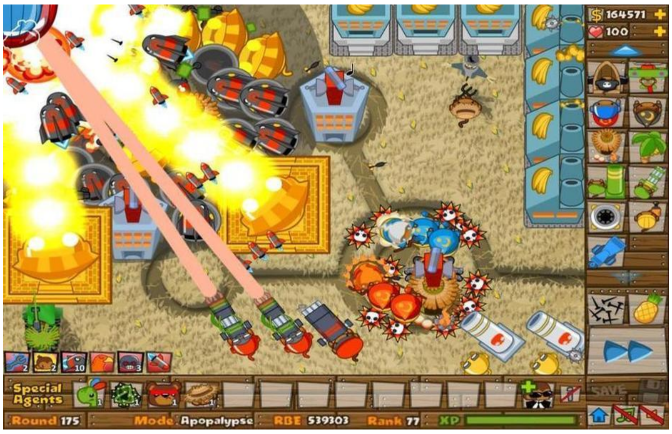
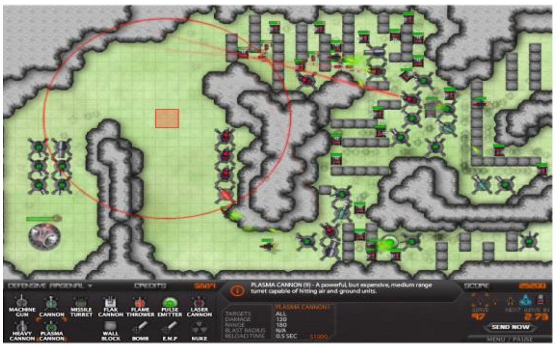


Figure 2.1: Bloons Tower Defense 5

2.3.2 Warzone

Warzone is tower defense game which contains high-technology weaponry and large open maps with different landscapes. The objective of this game is just to survive as long as possible during the game. There is one special features of this game that is it allows player to build wall to customize the map and fixed the path for the enemy. By building the strategic wall, enemies take longer time to reach the player base. Even though, the game had been design to have one kind of enemy which is air type enemy and not blocked by a wall and this make this is what makes the game fun and challenging to attract the player's passion. After the player has built these towers, it is essential that the player upgrades them, rather than buying more new ones. The enemy is free-path moving and player will be forced to think critically to build the tower at the right region.

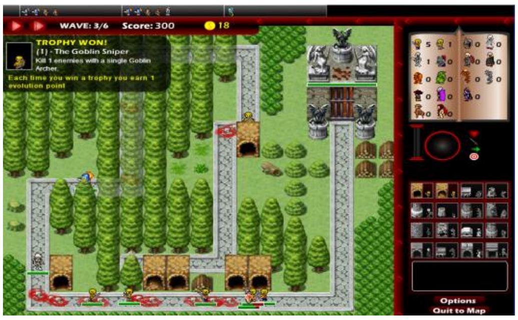


PLASMA CANNON (9) - A powerful, but expensive, medium range turret capable of hitting air and ground units.

Figure 2.2: Warzone Tower Defense Extended.

2.3.3 Demonic Guardians

Demonic Guardians is a straggly tower defense game with a superfluity of levels, enemies, and towers. Before starting the game, a hero – either ninja or knight will be selected by the player and it is controllable using the arrow key. Player should move the hero close to the action – they are going to do the most damage anyways. Player will be able to obtain trophies for doing in-game task if player is playing on either medium or hard. During unlock trophy moment, player will gain an evolution point and it can used to upgrade screen such unlock new tower, increase money gained and etc.



TROPHY WON!
(1) - The Goblin Sniper
Kill 1 enemies with a single Goblin Archer.
Each time you win a trophy you earn 1 evolution point

Figure 2.3: Demonic Guardians

2.3.4 Plant VS Zombie

Plant VS Zombie has five lanes to defend against the zombie and player is allowed to select six different plant types from armory at the beginning of each level for prevent player brain being eaten by zombie. In this game, each plant costs a certain amount of sunlight (money) to plant. Globs of sun will fall periodically to the earth, and player must click on them to add to owns supply before globs of sun disappear. Player also can earn additional sunlight by planting the sunflower. In each level of game, sunflower is one of the type of plant essential to the player because it will provide globs of sun to player in order to plant more expensive plant later. Next, this game also renders mini game, puzzle game, and etc to increase the players' passion.

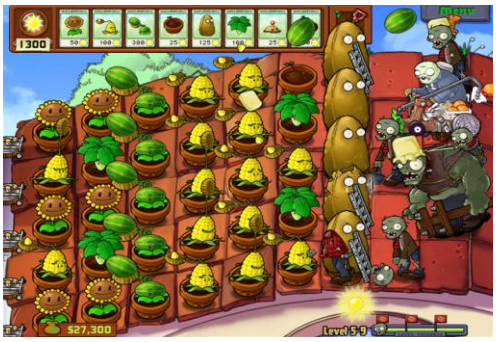


Figure 2.4: Plant VS Zombie

2.3.5 Born of Fire Tower Defense

Born of Fire Tower Defense is a game that combine with RTS game and role-playing game (RPG). This game is different with other TD game because of it is based on hero concept. There are four types of heroes which are demon incarnate, kitsune, king, and shield-maiden. Each character has its own hit-points (HP) and mana-points (MP). Heroes can up level after gaining some experiment in the game and improving their abilities after reached some level. Monsters that reach the base will bring damage to the hero but the hero can also heal themselves by using some specific skills. In addition, player can exchange two skills of each hero with other skills at any time. The choices of skill solely depend on the player strategy plan as long as the player can win the game.

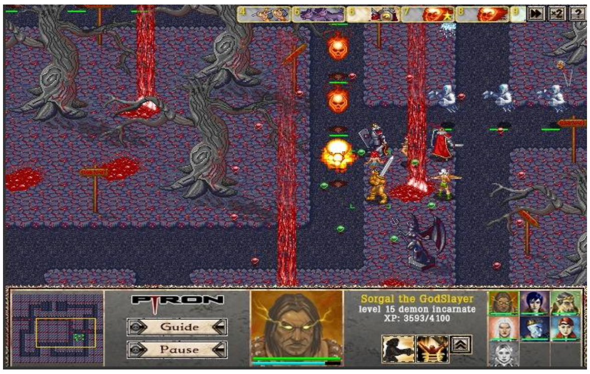


Figure 2.5: Born of Fire Tower Defense.

2.3.6 Integrated Defense

In this game, player is allowed to build cables, generators, and offensive towers. The goal for this game is to prevent the enemy from attacking the Center of Operations. The Center of Operations has a fixed life score and this game will end if life score

REFERENCES

Haibo W., Peter H. F. Ng*, Ben Niu and Simon C.K. Shiu. 2009. *Case Learning and Indexing in Real Time Strategy Games*. Fifth International Conference on Natural Computation.Hong Kong Polytechnic University.

Zanetti, S. and Rhalibi, A.E. 2004. *Machine learning techniques for FPS in Q3*. Proceedings of the 2004 ACM SIGCHI International Conference on Advances in computer entertainment technology, Singapore.

Cole, N., Louis, S.j., Miles, C. 2004. *Using a genetic algorithm to tune first-person shooter bots*. Evolutionary Computation.Congress on.Evolutionary Computation.

Louis, S.J. and Miles,C. 2005. *Playing to learn: case-injected genetic algorithms for learning to play computer games.*IEEE Transactions on Evolutionary Computation.

Hsieh, J.L. and Sun, C.T. 2008.*Building a player strategy model by analyzing replays of real-time strategy games.*Neural Networks.IJCNN.

Paul A. Rummell. 2011. *Adaptive AI to play tower defense game*. The 16th International Conference on Computer Games. Department of Computer Science, University of Victoria, Victoria, British Comlumbia, Canada.

Buro M. and Furtak T. 2003.*RTS game as test-bed for real-time research*. invited Paper at the Workshop on Game AI, JCIS.

Phillipa Avery, Julian Togelius, Elvis Alistar and Robert Pieter van Leeuwen, "Computational Intelligence and Tower Defense Games", Evolutionary Computation (CEC) IEEE Congress, 2011.

Back, T., Fogel, D.B. &Michalewicz, T. 2000Evolutionary Computation 1: Basic Algorithms and Operators. United Kingdom: Institute of Physics Publishing.

GuangmingLv, Xiaomeng Sun, Jian Wang. 2011. *A Simulated Annealing – New Genetic Algorithm and its Application.*International Conference on Electronics and Optoelectronics.Harbin Institute of Technology, China.

KyriakiGkoutioudi and Helen D. Karatza. 2012. *A Simulation Study of Multi-criteria Scheduling in Grid based on Genetic Algorithms*. 10th IEEE International Symposium on Parallel and Distributed Processing with Applications.Department of Informatics Aristotle University of Thessaloniki, Greece.

HongzeQiu, Wanli Zhou, and Hailong Wang. 2009. *A Genetic Algorithm-based Approach to Flexible Job-shop Scheduling Problem.*Fifth International Conference on Natural Computation. Shandong University, China.

Lee KM, Yamakawa T, Lee KM. 1998. *A Genetic Algorithm for General Machine Scheduling Problems.*International Journal of Knowledge-Based Electronic.

Tai-shan Yan, Yong-qing Tao, and Du-wu Cui. 2007. *Research on Handwritten Numeral Recognition Method Based on Improved Genetic Algorithm and Neural Network*. International Conference on Wavelet Analysis and Pattern Recognition.Xi'an University of Technology, China.

PengfeiGuo, Xuezhi Wang, and Yingshi Han. 2010. *The Enhanced Genetic Algorithms for the Optimization Design*. Third International Conference on Biomedical Engineering and Informatics.Liaoning University of Technology, China.

RushilRaghavjee and NelishiaPillay. 2012. *A Comparison of Genetic Algorithms and Genetic Programming in Solving the School Timetabling Problem*. Fourth Worth Congress on Nature and Biologically Inspired Computing(NaBIC). University of KwaZulu-Natal Pietermaritzburg, South Africa.

KorneelDuyvesteyen and UzayKaymak. 2005. *Genetic Programming in Economic Modeling.* Erasmus University Rotterdam, The Netherlands.

Michael D.Kramer and Du Zhang. 2000. *GAPS: a Genetic Programming System*. Department of Computer Science California State University Sacramento, U.S.A.

PavelPetr, Jiri Krupka, and RomanaProvaznikova. *Mathematics Model Design Based on Genetic Programming*. University of Pardubice, CzehPepublic.

Hui-Hua Yang, Shih-Huang Chen, Jui-Ying Hung, Ching-Tsung Hung, and Meng-Lung Chung. 2010. *Utilization of Genetic Programming to Establish Demand Forecast in Taiwan International Flights*. Toko University, Taiwan.

Kumar Abhishek, Abhay Kumar, Rajeev Ranjan, and Sarthak Kumar. 2012. *A Rainfall Prediction Model using Artificial Neural Network*. IEEE Control and System Graduate Reseach Colloquium.NIT Patna-800005 and Deolitte India, Pune.

Mr. P. A. Kharat and Dr. S. V. Dudul. 2011. *Clinical Decision Support System Based On Jordan/Elman Neural Network*. Department of Information Technology (Maharashta), India. Amravati University, India.

Hee-Heon Song, Sun-Mee Kang, and Seong-Whan Lee. 1996. *A New Recurrent Neural Network Architecture for Pattern Recognition*. IEEE Proceedings of ICPR.Yusong-ku Taejon and Seongbuk-ku, Korea.

Rahul P. Deshmukh and A. A. Ghatol. 2010. *Comparative study of Jordan and Elman model of neural network for short term flood forecasting*. IEEE.Indian Institute of Technology, Bombay Powai and Dr. BabasahebAmbedkar Technological University, India.

Seong-Whan Lee and Young-Joon Kim. 1995. *A New Type of Recurrent Neural Network for Handwritten Character Recognition.* IEEE.Department of

Computer Science Korea University and Technical Laboratory BIT Computer Company, Korea.

Vaibhav Narayan chunekar and Hemant P. Ambulgekar. 2009. *Approach of Neural Network to Diagnose Breast Cancer on three different Data Set*. International Conference on Advances in Recent Technologies in Communication and Computing.Computer Department VJTI, India.

HarshaniR.K.Nagahamulla, UdithaR.Ratnayake and AsangaRatnaweera. 2012*. An Ensemble of Artificial Neural Networks in Rainfall Forecasting*. The International Conference on Advances in ICT for Emerging Regions. Wayamba University of Sri Lanka and University of Peradeniya, Sri Lanka.

Shuyan Chen, Wei Wang, GaofengQu, and Jian Lu. 2007. *Application of Neural Network Ensembles to Incident Detection*. IEEE International Conference on Integration Technology.Southeast University and Nanjing University of Posts and Telecommunications, China.

KhobaibZaamout and John Z. Zhang. 2012. *Improving Classification through Ensemble Neural Networks*. International Conference on Natural Computation. University of Lethbridge, Canada.

Mohd.Haris Lye b. Abdullah, V. Ganapathy. 2000. *Neural Network Ensemble for Financial Trend Prediction*. Multimedia University, Malaysia.

Jian Wang, Jingfeng Yang, Shaoofa LI, Qiufang Dai, JianxingXie. 2007. *Number Image Recognition Based on Neural Network Ensemble*. Third International Conference on Natural computation. South China Agriculture University and South China University of Technology, China.

Eiben, A.E. & Smith, J.E. 2003. Introduction to Evolutionary Computing. Netherlands: Springer.

Asela Gunawardana and Guy Shani. 2009. *A Survey of Accuracy Evaluation Metrics of Recommendation Tasks*. Journal of Machine Learning Research 10. Microsoft Research One Microsoft Way Redmond, USA and Information Systems Engineering Ben Gurion University Beer Sheva, Israel.

Munroe, D.R. (2004) Genetic Programming: The ratio of Crossover to Mutation as a function of time.

Chang, K.T et al. (2011) Evolving Neural Controllers using GA for Wacraft 3-Reaal Time Strategy Game.

Munroe, D.R. (2004) Genetic Programming: The ratio of Crossover to Mutation as a function of time.

Jeff Heaton. Introduction to Neural Network with Java, Second Edition (page 159). Heaton Research, Inc.

Imran Maqsood, Muhammad Riaz Khan, Ajith Abraham. 2004. An Ensemble of Neural Networks for Weather Forecasting. Neural Comput & Applic. Faculty of Engineering, University of Regina, Regina, Canada.

Tse Guan Tan, Anthony, P., Teo, J., and Jia Hui Ong. 2011. Neural Network Ensembles For Video Game AI Using Evolutionary Multi-Objective Optimization. International Conference on Hybrid Intelligent Systems. 2011. 605-610.

Pulido, M., Melin, P., and Castillo, O. 2014. Optimization of Ensemble Neural Networks with Fuzzy Integration Using the Particle Swarm Algorithm for the US Dollar/MX Time Series Prediction. IEEE Conference on Norbert Wiener in the 21st Century. 2014. 1-7.