HYBRID ARTIFICIAL BEE COLONY AND FLOWER POLLINATION ALGORITHM FOR GRID-BASED OPTIMAL PATHFINDING

AIMI NAJWA BINTI SABRI

A thesis submitted in partial fulfilment of the
requirements for the award of degree of
Master of Philosophy

School of Computing
Faculty of Engineering
Universiti Teknologi Malaysia

JANUARY 2019

ACKNOWLEDGEMENT

First of all, I would like to express my praise to Allah as I have given a chance to complete my research. My highest appretiation to my husband, Syed Muhammad Syahmi for his support and attention along my journey to finish this research. Thank you to my supervisors, Dr Nor Haizan Mohamed Radzi, Prof Dr Habibollah Haron and Dr Haswadi Hassan for their guidance, kindness and support encourage me during this journey.

Thank you to my family and family in law for their support and understanding while I am trying to finish my work. Thank you for the kindness and morale support encourage me to continue this project with high determination and spirit. Not to forget, I would also like to say thank you to my friends especially Faizah, Fatin, Dhabitah, Yana and Nasir. Thank you for the great opinions and help so that I could solve the problems. Thanks for the kind words whenever I feel stressed in doing this research.

Last but not least, for everybody whom I do not mention their name. I appreciate your help and support too. I hope that my research will benefit others and become useful reference in the future.

ABSTRACT

Pathfinding is essential and necessary for agent movement used in computer games and many other applications. Generally, the pathfinding algorithm searches the feasible shortest path from start to end locations. This task is computationally expensive and consumes large memory, particularly in a large map size. Obstacle avoidance in the game environment increases the complexity to find a new path in the search space. A huge number of algorithms, including heuristic and metaheuristics approaches, have been proposed to overcome the pathfinding problem. Artificial Bee Colony (ABC) is a metaheuristic algorithm that is robust, has fast convergence, high flexibility, and fewer control parameters. However, the best solution founded by the onlooker bee in the presence of constraints is still insufficient and not always satisfactory. A number of variant ABC algorithms have been proposed to achieve the optimal solution. However, it is difficult to simultaneously achieve the optimal solution. Alternatively, Flower Pollination Algorithm (FPA) is one of promising algorithms in optimising problems. The algorithm is easier to implement and faster to reach an optimum solution. Thus, this research proposed Artificial Bee Colony – Flower Pollination Algorithm to solve the pathfinding problem in games, in terms of path cost, computing time, and memory. The result showed that ABC-FPA improved the path cost result by 81.68% and reduced time by 97.84% as compared to the ABC algorithm, which led to a better pathfinding result. This performance indicated that ABC-FPA pathfinding gave better quality pathfinding results.

ABSTRAK

Pencarian laluan ialah satu algoritma yang penting dan diperlukan dalam pergerakan agen yang digunakan dalam permainan komputer dan lain-lain aplikasi. Secara umumnya, algoritma pencarian laluan bertujuan untuk mencari laluan terpendek yang sesuai dari titik permulaan permainan hingga akhir. Pencarian laluan ini dikira mahal dan menggunakan ingatan yang besar, terutamanya dalam saiz peta yang besar. Strategi mengelak halangan dalam persekitaran permainan meningkatkan kerumitan untuk mencari laluan baru dalam ruang carian. Sebilangan besar algoritma termasuk pendekatan heuristik dan metaheuristik telah dicadangkan untuk mengatasi masalah pencarian laluan. *Artificial Bee Colony* (ABC) merupakan algoritma metaheuristik yang teguh, mempunyai penumpuan yang cepat, fleksibiliti yang tinggi, dan parameter kawalan yang kurang. Walau bagaimanapun, penyelesaian terbaik yang diasaskan oleh lebah pemerhati (*onlooker bee*) dalam persekitaran yang ada kekangan masih tidak mencukupi dan tidak memuaskan.Pelbagai algoritma ABC telah dicadangkan untuk mencapai penyelesaian optimum. Namun, penyelesaian yang optimum masih sukar dicapai. Sebagai alternatif, *Flower Pollination Algorithm* (FPA) adalah salah satu algoritma bagi mengoptimumkan masalah. Algoritma ini lebih mudah untuk dilaksanakan dan lebih cepat mencapai penyelesaian optimum. Oleh itu, kajian ini telah mencadangkan *Artificial Bee Colony* (ABC) – *Flower Pollination Algorithm* (FPA) untuk menyelesaikan masalah pencarian laluan dalam permainan komputer dari segi kos laluan, masa pengiraan dan ingatan. Hasil kajian menunjukkan bahawa ABC-FPA telah meningkatkan hasil kos laluan sebanyak 81.68% dan mengurangkan masa sebanyak 97.84% berbanding dengan algoritma ABC. Algoritma cadangan ini membawa kepada hasil laluan yang lebih baik. Prestasi ini menunjukkan bahawa pencarian laluan ABC-FPA telah memberikan kualiti yang lebih baik.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

2D	-	Two Dimensional
3D	-	Three Dimensional
ABC	-	Artificial Bee Colony
ACO	-	Ant Colony Optimisation
AI	-	Artificial Intelligence
BA	-	Bee Algorithm
BFS	-	Breadth First Search
CLPSO	-	Comprehensive Learning Particle Swarm Optimisation
CPU	-	Central Processing Unit
DFS	-	Depth First Search
FPA	-	Flower Pollination Algorithm
FSO	-	Fish Swarm Optimisation
GA	-	Genetic Algorithm
GABC	-	Gbest-guided Artificial Bee Colony
GABCS	-	Global Artificial Bee Colony Search
HABC	-	Hierarchical Artificial Bee Colony
HPA*	-	Hierarchical Pathfinding A*
IDA*	-	Iterative Deepening A*
MFPA	-	Multi-Objective Flower Pollination Algorithm
PSO	-	Particle Swarm Optimisation
SFPA	-	Single Objective Flower Pollination Algorithm
SI	-	Swarm Intelligence
TSP	-	Travelling Salesman Problem

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 Introduction

This chapter is an overview of the research conducted. Problem background, problem statement, aim of the research study, objectives of research study, and scopes of research study and the significant of study are discussed in this chapter.

1.2 Research Overview

Pathfinding is an indispensable part in many domains for instance video games, robotic and global positioning system (GPS) (Algfoor et al., 2015). Fundamentally, the similar issues in using pathfinding in these domains are to find a traversable path for a unit or agent from the starting location to the end location. However, different application may address different constraints and requirements of pathfinding problem. In GPS application for example, pathfinding problem seeks for the most safest and shortest path to the goal. On the other hand, in robotic or video games, apart from finding the shortest path, obstacles and optimal path are often placed as constraints and requirements of the pathfinding problem as shown in Figure 1.1. This has resulted the pathfinding algorithm prioritizes different value and appears in many forms and sizes.

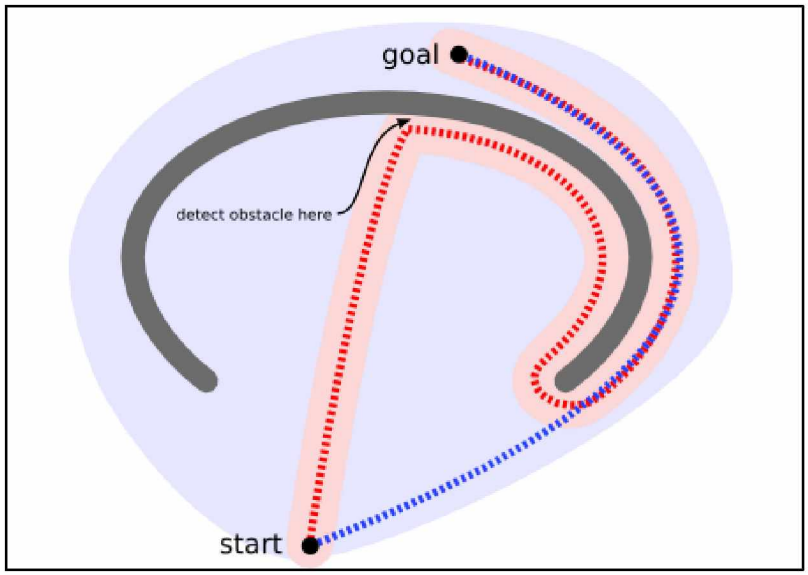

Figure 1.1: An illustration of pathfinding in games (http://theory.stanford.edu/~amitp, 1997).

Generally, there are two significant steps in pathfinding which are the graph generation and a pathfinding algorithm (Algfoor et al., 2015). The complexity of the graph generation much depends on the terrain topology, for example 2D or 3D topology and the base of the layout or map of the game world. The graph generation provides a search space to plan a path to any destination via pathfinding algorithm. The graph usually consists of nodes that represent possible traversal points available in such game world. The nodes will be kept in the memory during the generation of the path. Simple search space will indicate the least number of nodes in the graph to be visited by the pathfinding algorithm. Therefore, the pathfinding algorithm runs faster in simple search space since least nodes to be inspected and less memory to be used to keep these nodes. Thus, the performance of pathfinding algorithm has a correlation to the graph generation. There are various ways to represent a search space including a rectangular grid, quad tree, convex polygon, point of visibility and generalised cylinder. These various search spaces contributes to the complexity processes in finding the path (Mehta et al., 2015.) In today's fast paced games world, the environment becomes more complex, regardless of the map size and the number of units (Algfoor et al., 2015; Graham et al., 2015). Large size of game environment will require more searching time for the desirable path and consume large of memory which has further increased the complexity of problem in finding the path.

The existence of obstacles and dynamic games environment increase the complexity of pathfinding problem and contribute to the performance of the algorithm. Agents need to seek the optimal path while avoiding the obstacles (Cui et al., 2011; Botea et al., 2013; Mathew et al., 2015; Barnouti et al., 2016). Most of the games are inefficient to handle dynamic environment, which lead to consume large of memory and CPU resources (Graham et al., 2003; Björnsson et al., 2006; Bulitko et al., 2011; Lawrence et al., 2013). Games in dynamic environment are computationally expensive to calculate the new path. The sophistication of games was limited due to the time required to find a path (Graham et al., 2004; Bulitko et al., 2011; Kelman, 2017).

Therefore, over the past decades researcher include optimization approach in pathfinding algorithm to deal with the increasing trend of games complexity. This thesis attempt to develop a hybrid pathfinding algorithm by exploiting of the ABC and FPA algorithm potential to solve to optimization problem, to generate a shortest path in 2D games environment.

1.3 Problem Background

The interaction between game environment and search process are important in pathfinding games. An overlay map with cost information is required (Davis, 2000; Tozour, 2002; Champandard, 2009; Cui et al., 2011). There are many ways to differentiate type of search graph such as navigation mesh, waypoint and grid-based graph. Each map representation has its own advantages and disadvantages. Typically, grid-based environment is used to speed up the pathing search (Harabor, 2014; Andreychuk et al., 2018). It is easy to understand and apply to the map. Thus, the simpler graph makes memory become more efficient (only one bit of storage is required per grid cell), but it is not guarantee that the cost is accurately optimal. However, the distance of the grid is small, it may need many steps to be taken before it reaches the target.

Generally all pathfinding algorithm aims to minimize the cost and find the shortest path from the start to the end point. One of the earliest pathfinding algorithm

and become the base of many pathfinding algorithm is Dijkstra. Dijkstra works by visiting and evaluating the nodes in the graph one after another from the initial point. Thus, it usually end up with exploring all nodes and make the searching process become more slower since extra time and memory required to inspects all the nodes (Lester, 2005; Cui et al., 2011; Vinther et al., 2015).To improve the situation, researcher has introduced A* algorithm which is an extension of Dijkstra but it includes heuristic function that estimates the cost to the goal upon inspection the nodes in the graph. A* algorithm is one of the most renowned pathfinding algorithm used in video games (Björnsson et al., 2006; Xu, 2011; Bulitko et al., 2011; Mathew et al., 2015; Barnouti et al, 2016; Kelman, 2017). However, it also has it's drawback since A* is not always optimal in dynamic environment due to excessive memory usage that lead to inefficient computational time (Cazenave, 2006; Mapaila, 2012; Algfoor et al., 2015; Hagelbäck, 2016).

Extensive work has been done to modify A* algorithm to speed up search through better memory organization and execution of time (Botea et al., 2004; Lester, 2005; Björnsson et al. 2006; Cazenave, 2006; Xiau, 2011; Mapaila, 2012; Reddy, 2013). As for instance, Jump Point Search (JPS) managed to reduce number of explored nodes but the running time is not improved over A* (Lester, 2005). Sturtevant (2005) has combining hierarchical abstraction with fast partial path refinement in Partial Refinement A* (PRA*) by mapping small connected regions. The process decreases the number of nodes by doing multiple refinements. Hierarchical Pathfinding A* (HPA*) and PRA* are combined to decrease the memory used for abstraction (Sturtevant, 2007). However, these variants of A* stored nodes information and end up uses more memory and not efficient at all (Botea et al., 2004; Sturtevant, 2007; Halldórsson, 2015). Leigh et al, developed hybrid A* with GA and expected to outperform standard A* (Leigh et al., 2007). Yet, it resulted more expanded nodes to be needed thus slower the performances of A* and the different in win ratio is small for simple map environment (Petereit, 2012; Hagelbäck, 2012; Hagelbäck, 2016). In short, the current study carried by the researcher generally attempt to optimize the use of memory and reducing the time by improving the number of visited nodes in the graph.

Metaheuristic algorithm has become popular due to its success in obtaining optimal solutions for complex problem in engineering and industry (Salnon et al., 2006; Mocholi et al., 2010; Mora et al., 2012; Mohanty et al., 2013; Koceski et al., 2014;Pandey et al. 2017; Saad et al., 2017), especially in games and robotic. Generally, metaheuristic improves the search strategy through random and multi-agent approach search strategies. In recent years, several new search metaheuristic have been proposed including Artificial Bee Colony (Karaboga et al., 2007), Firefly Algorithm (Yang, 2008), Cuckoo Search (Yang et al., 2009), Bat Inspired Algorithm (Yang, 2010) and Flower Pollination Algorithm (Yang, 2012). These algorithms have potential to solve complex problem especially when the number of iterations are limited. They also have shared many similarities in general. As for instance, they pick a set of random solution and operate a fitness value to evaluate the population. They update the population and randomly search for the optimum. The sharing information mechanism is used to find the best solution. However, among these algorithm, ABC was chosen because of its criteria that significant to the current research which help in optimizing the resources and obtain the shortest path in pathfinding games.

Recently, Artificial Bee Colony (ABC) is seen to be more competitive to others metaheuristic algorithm in term of optimization performance. Due to its high efficiency, many development and improvement have made to be utilized for different purposes (Zhu et al., 2010; Banharnsakun et al., 2011; Gao et al., 2012; Li et al., 2012; Xiang et al., 2013; Garg et al., 2013; Li et al., 2016). ABC is better than others metaheuristic algorithm in many problems because of strong robustness, fast convergence and high flexibility and fewer control parameters (Karaboga, 2007; Yan, 2011; Saad et al., 2017). However, the best solution founded by onlooker bee in presence of constraints is still insufficient and not always satisfactory (Mezura-Montes et al., 2010; Yan, 2011; Stanarevic et al., 2011; Banharnsakun et al., 2011).

In relation to that, Flower Pollination Algorithm (FPA) is more convenient in optimizing the solution in terms of time and memory. FPA is one of the nature-inspired algorithms developed by Yang et al. (2012) for engineering tasks. The efficiency of this algorithm is proven to be outperforming even for the evolutionary based algorithms (Prathiba et al., 2014; Yang et al, 2014; Abdel-Raouf et al., 2014; Wang et al., 2014; Łukasik et al., 2015). The algorithm with less number of operators can be

easily implemented in any algorithm. Since FPA is simple and easy, the outcome was very efficient with almost exponential convergence rate. Therefore, in order to improve the accuracy of optimal solution and obtain the fine convergence ability of ABC, FPA search method was recommended to solve this problem.

1.4 Problem Statement

Researchers in the past had developed various solutions in tackling pathfinding problems and provide optimal solution with regard to the given constraints. Based on the background of the problem, the improvement aspect is mainly focuses on using the new approach by implementing FPA into ABC pathfinding in games. The focuses are indicated as below:

FPA helps ABC in finding optimal path for pathfinding in 2D dynamic game environment.
Implementation of FPA is used to improve computational time and memory.

One of the significant approaches to handle the pathfinding problem is through the implementation of hybrid algorithm. The proposed algorithm exploits good properties of Artificial Bee Colony (ABC) algorithm and Flower Pollination algorithm (FPA) in order to develop a new algorithm which is more efficient than the original algorithms. Hence, this study, proposed ABC algorithm by implementing FPA to find the optimal path pathfinding in grid-based games environment.

From reviewing the problem statement of this research study, three research questions were highlighted as follows:

1 How to obtain the optimal solution in pathfinding problem?
2 How can ABC and FPA be hybrid and get the optimal solution?
3 Can this hybrid ABC-FPA for pathfinding solve the pathfinding problem?

1.5 Aim

This study aims to develop hybrid algorithm to obtain optimal path in grid-based games environment.

1.6 Objective

This study highlights the three main objectives in order to answer the research questions, which are:

4 To develop pathfinding in games using ABC algorithm.
5 To propose hybrid ABC-FPA pathfinding for optimal path.
6 To validate the proposed solution of time complexity, memory and best cost of the hybrid ABC-FPA.

1.7 Scope

The limitation of this research study are listed as follows:

1 This study only focuses on single player with 2D movement.
2 The obstacles are placed randomly in grid environment.
3 This study will be conducted by using MATLAB R2013a.
4 The size maps are setup in 200x200, 500x500 and 1000x1000.

REFERENCES

Algfoor, Z. A., Sunar, M. S., & Kolivand, H. 2015.. A comprehensive study on pathfinding techniques for robotics and video games. *International Journal of Computer Games Technology*, *2015*, 7.

Graham, R., McCabe, H. and Sheridan, S., 2003. Pathfinding in computer games. *The ITB Journal*, *4*(2), p.6.

Graham, R., McCabe, H. and Sheridan, S., 2004. Neural networks for real-time pathfinding in computer games. *The ITB Journal*, *5*(1), p.21.

Björnsson, Y. and Halldórsson, K., 2006. Improved Heuristics for Optimal Path-finding on Game Maps. *AIIDE*, *6*, pp.9-14.

Bulitko, V., Björnsson, Y., Sturtevant, N.R. and Lawrence, R., 2011. Real-time heuristic search for pathfinding in video games. In *Artificial Intelligence for Computer Games* (pp. 1-30). Springer, New York, NY.

Cui, X. and Shi, H., 2011. A*-based pathfinding in modern computer games. *International Journal of Computer Science and Network Security*, *11*(1), pp.125-130.

Botea, A., Bouzy, B., Buro, M., Bauckhage, C. and Nau, D., 2013. Pathfinding in games. In *Dagstuhl Follow-Ups* (Vol. 6). Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik.

Kelman, A., 2017. Utilizing Swarm Intelligence Algorithms for Pathfinding in Games.

Yannakakis, G.N. and Togelius, J., 2018. *Artificial Intelligence and Games*. New York: Springer.

Mathew, G.E. and Malathy, G., 2015, February. Direction based heuristic for pathfinding in video games. In *Electronics and Communication Systems (ICECS), 2015 2nd International Conference on* (pp. 1651-1657). IEEE.

Barnouti, N.H., Al-Dabbagh, S.S.M. and Naser, M.A.S., 2016. Pathfinding in Strategy Games and Maze Solving Using A* Search Algorithm. *Journal of Computer and Communications*, *4*(11), p.15.

Lawrence, R. and Bulitko, V., 2013. Database-driven real-time heuristic search in

video-game pathfinding. *IEEE Transactions on Computational Intelligence and AI in Games*, *5*(3), pp.227-241.

Hagelbäck, J., 2016. Hybrid Pathfinding in StarCraft. *IEEE Trans. Comput. Intellig. and AI in Games*, *8*(4), pp.319-324.

Andreychuk, A. and Yakovlev, K., 2018, September. Path Finding for the Coalition of Co-operative Agents Acting in the Environment with Destructible Obstacles. In *International Conference on Interactive Collaborative Robotics* (pp. 13-22). Springer, Cham.

Cazenave, T., 2006, May. Optimizations of data structures, heuristics and algorithms for path-finding on maps. In *Computational Intelligence and Games, 2006 IEEE Symposium on* (pp. 27-33). IEEE.

Sturtevant, N. and Buro, M., 2005, July. Partial pathfinding using map abstraction and refinement. In *AAAI* (Vol. 5, pp. 1392-1397).

Sturtevant, N.R., 2007. Memory-Efficient Abstractions for Pathfinding. *AIIDE*, *684*, pp.31-36.

Vinther, A.S.H., Vinther, M.S.H. and Afshani, P., 2015. Pathfinding in Two-dimensional Worlds. *no. June*.

Lester, P., 2005. A* pathfinding for beginners. *online]. GameDev WebSite. http://www. gamedev. net/reference/articles/article2003. asp (Acesso em 08/02/2009)*.

Botea, A., Müller, M. and Schaeffer, J., 2004. Near optimal hierarchical path-finding. *Journal of game development*, *1*(1), pp.7-28.

Reddy, H., 2013. PATH FINDING-Dijkstra's and A* Algorithm's. *International Journal in IT and Engineering*, pp.1-15.

Halldórsson, K., 2015. *Using Map Decomposition to Improve Pathfinding* (Doctoral dissertation).

Petereit, J., Emter, T., Frey, C.W., Kopfstedt, T. and Beutel, A., 2012, May. Application of Hybrid A* to an autonomous mobile robot for path planning in unstructured outdoor environments. In *Robotics; Proceedings of ROBOTIK 2012; 7th German Conference on* (pp. 1-6). VDE.

Hagelbäck, J., 2012. *Multi-agent potential field based architectures for real-time strategy game bots* (Doctoral dissertation, Blekinge Institute of Technology).

Solnon, C. and Bridge, D., 2006. An ant colony optimization meta-heuristic for subset selection problems. *System engineering using particle swarm optimization*, pp.7-

29.

Mohanty, P.K. and Parhi, D.R., 2013, December. Cuckoo search algorithm for the mobile robot navigation. In *International Conference on Swarm, Evolutionary, and Memetic Computing* (pp. 527-536). Springer, Cham.

Pandey, A. and Parhi, D.R., 2017. Optimum path planning of mobile robot in unknown static and dynamic environments using Fuzzy-Wind Driven Optimization algorithm. *Defence Technology*, *13*(1), pp.47-58.

Koceski, S., Panov, S., Koceska, N., Zobel, P.B. and Durante, F., 2014. A novel quad harmony search algorithm for grid-based path finding. *International Journal of Advanced Robotic Systems*, *11*(9), p.144.

Mocholi, J.A., Jaen, J., Catala, A. and Navarro, E., 2010. An emotionally biased ant colony algorithm for pathfinding in games. *Expert Systems with Applications*, *37*(7), pp.4921-4927.

Mora, A.M., Fernández-Ares, A., Merelo, J.J., García-Sánchez, P. and Fernandes, C.M., 2012. Effect of noisy fitness in real-time strategy games player behaviour optimisation using evolutionary algorithms. *Journal of Computer Science and Technology*, *27*(5), pp.1007-1023.

Karaboga, D. and Basturk, B., 2007, June. Artificial bee colony (ABC) optimization algorithm for solving constrained optimization problems. In *International fuzzy systems association world congress* (pp. 789-798). Springer, Berlin, Heidelberg.

Yang, X.S., 2010. Firefly algorithm, stochastic test functions and design optimisation. *arXiv preprint arXiv:1003.1409*.

Yang, X.S. and Deb, S., 2009, December. Cuckoo search via Lévy flights. In *Nature & Biologically Inspired Computing, 2009. NaBIC 2009. World Congress on* (pp. 210-214). IEEE.

Yang, X.S., 2012, September. Flower pollination algorithm for global optimization. In *International conference on unconventional computing and natural computation* (pp. 240-249). Springer, Berlin, Heidelberg.

Saad, A.E.H., Dong, Z. and Karimi, M., 2017. A Comparative Study on Recently-Introduced Nature-Based Global Optimization Methods in Complex Mechanical System Design. *Algorithms*, *10*(4), p.120.

Gao, W.F. and Liu, S.Y., 2012. A modified artificial bee colony algorithm. *Computers & Operations Research*, *39*(3), pp.687-697.

Zhu, G. and Kwong, S., 2010. Gbest-guided artificial bee colony algorithm for

numerical function optimization. *Applied mathematics and computation*, *217*(7), pp.3166-3173.

Li, G., Niu, P. and Xiao, X., 2012. Development and investigation of efficient artificial bee colony algorithm for numerical function optimization. *Applied soft computing*, *12*(1), pp.320-332.

Banharnsakun, A., Achalakul, T. and Sirinaovakul, B., 2011. The best-so-far selection in artificial bee colony algorithm. *Applied Soft Computing*, *11*(2), pp.2888-2901.

Xiang, W.L. and An, M.Q., 2013. An efficient and robust artificial bee colony algorithm for numerical optimization. *Computers & Operations Research*, *40*(5), pp.1256-1265.

Li, X. and Yang, G., 2016. Artificial bee colony algorithm with memory. *Applied Soft Computing*, *41*, pp.362-372.

Stanarevic, N., Tuba, M. and Bacanin, N., 2011. Modified artificial bee colony algorithm for constrained problems optimization. *International journal of mathematical models and methods in applied sciences*, *5*(3), pp.644-651.

Mezura-Montes, E., Damián-Araoz, M. and Cetina-Domingez, O., 2010, July. Smart flight and dynamic tolerances in the artificial bee colony for constrained optimization. In *Evolutionary Computation (CEC), 2010 IEEE Congress on*(pp. 1-8). IEEE.

Yang, X.S., Karamanoglu, M. and He, X., 2014. Flower pollination algorithm: a novel approach for multiobjective optimization. *Engineering Optimization*, *46*(9), pp.1222-1237.

Abdel-Raouf, O. and Abdel-Baset, M., 2014. A new hybrid flower pollination algorithm for solving constrained global optimization problems. *International Journal of Applied Operational Research-An Open Access Journal*, *4*(2), pp.1-13.

Wang, R. and Zhou, Y., 2014. Flower pollination algorithm with dimension by dimension improvement. *Mathematical Problems in Engineering*, *2014*.

Łukasik, S. and Kowalski, P.A., 2015. Study of flower pollination algorithm for continuous optimization. In *Intelligent Systems' 2014* (pp. 451-459). Springer, Cham.

Harabor, D., 2014. Fast and Optimal Pathfinding.

Reuderink, B., Nijholt, A. and Poel, M., 2009, June. Affective Pacman: A frustrating game for brain-computer interface experiments. In *International conference on*

intelligent technologies for interactive entertainment (pp. 221-227). Springer, Berlin, Heidelberg.

Khenissi, M.A., Essalmi, F. and Jemni, M., 2013, October. A learning version of Pacman game. In *Information and Communication Technology and Accessibility (ICTA), 2013 Fourth International Conference on* (pp. 1-3). IEEE.

Anguelov, B., 2011. *Video game pathfinding and improvements to discrete search on grid-based maps*(Doctoral dissertation, University of Pretoria).

Sturtevant, N.R., 2012. Benchmarks for grid-based pathfinding. *IEEE Transactions on Computational Intelligence and AI in Games*, *4*(2), pp.144-148.

Yap, P., 2002, May. Grid-based path-finding. In *Conference of the Canadian Society for Computational Studies of Intelligence* (pp. 44-55). Springer, Berlin, Heidelberg.

Cui, X. and Shi, H., 2011. Direction oriented pathfinding in video games. *International Journal of Artificial Intelligence & Applications*, *2*(4), p.1.

Jong, D., Kwon, I., Goo, D. and Lee, D., 2015, November. Safe Pathfinding Using Abstract Hierarchical Graph and Influence Map. In *Tools with Artificial Intelligence (ICTAI), 2015 IEEE 27th International Conference on* (pp. 860-865). IEEE.

Aridhi, S., Lacomme, P., Ren, L. and Vincent, B., 2015. A MapReduce-based approach for shortest path problem in large-scale networks. *Engineering Applications of Artificial Intelligence*, *41*, pp.151-165.

Björnsson, Y., Enzenberger, M., Holte, R.C. and Schaeffer, J., 2005. Fringe Search: Beating A* at Pathfinding on Game Maps. *CIG*, *5*, pp.125-132.

Potts, C.M. and Krebsbach, K.D., 2012, March. Iterative-Expansion A. In *FLAIRS Conference*.

Sarkar, U.K., Chakrabarti, P.P., Ghose, S. and De Sarkar, S.C., 1991. Reducing reexpansions in iterative-deepening search by controlling cutoff bounds. *Artificial Intelligence*, *50*(2), pp.207-221.

Zhou, R. and Hansen, E.A., 2006. Breadth-first heuristic search. *Artificial Intelligence*, *170*(4), pp.385-408.

Lippi, M., Ernandes, M. and Felner, A., 2012. Efficient Single Frontier Bidirectional Search. In *SOCS*.

Barker, J.K. and Korf, R.E., 2015, January. Limitations of Front-To-End Bidirectional Heuristic Search. In *AAAI* (pp. 1086-1092).

Zahavi, U., Felner, A., Burch, N. and Holte, R.C., 2010. Predicting the performance of IDA* using conditional distributions. *Journal of Artificial Intelligence Research*, *37*, pp.41-83.

Vermette, J., 2011. A Survey of Path-finding Algorithms Employing Automatic Hierarchical Abstraction. *Journal of the Association for Computing Machinery*, *377*, p.383.

Leigh, R., Louis, S.J. and Miles, C., 2007, April. Using a genetic algorithm to explore A*-like pathfinding algorithms. In *Computational Intelligence and Games, 2007. CIG 2007. IEEE Symposium on* (pp. 72-79). IEEE.

Mehta, P., Shah, H., Shukla, S. and Verma, S., A Review on Algorithms for Pathfinding in Computer Games.

Shah, H., Ghazali, R. and Hassim, Y.M.M., 2014. Honey bees inspired learning algorithm: nature intelligence can predict natural disaster. In *Recent Advances on Soft Computing and Data Mining* (pp. 215-225). Springer, Cham.

Zhang, Y. and Wu, L., 2012. Artificial bee colony for two dimensional protein folding. *Advances in Electrical Engineering Systems*, *1*(1), pp.19-23.

Mohamed, A.F., Elarini, M.M. and Othman, A.M., 2014. A new technique based on Artificial Bee Colony Algorithm for optimal sizing of stand-alone photovoltaic system. *Journal of advanced research*, *5*(3), pp.397-408.

Wang, S., Yang, J., Liu, G., Du, S. and Yan, J., 2016. Multi-objective path finding in stochastic networks using a biogeography-based optimization method. *Simulation*, *92*(7), pp.637-647.

Nabil, E., 2016. A modified flower pollination algorithm for global optimization. *Expert Systems with Applications*, *57*, pp.192-203.

Zhang, A., Li, C. and Bi, W., 2016. Rectangle expansion A∗ pathfinding for grid maps. *Chinese Journal of Aeronautics*, *29*(5), pp.1385-1396.

Singh, S. and Pathak, V., 2014. Fitness based mutation in artificial bee colony algorithm. In *Int. J. Comput. Appl. National Seminar on Recent Advances in Wireless Networks and Communications* (pp. 26-30).

Watanabe, Y., Takaya, M. and Yamamura, A., 2015. Fitness function in ABC algorithm for uncapacitated facility location problem. In *Information and Communication Technology* (pp. 129-138). Springer, Cham.

Le Dinh, L., Vo Ngoc, D. and Vasant, P., 2013. Artificial bee colony algorithm for solving optimal power flow problem. *The Scientific World Journal*, *2013*.

Binitha, S. and Sathya, S.S., 2012. A survey of bio inspired optimization algorithms. *International Journal of Soft Computing and Engineering*, *2*(2), pp.137-151.

Karaboga, D. and Akay, B., 2009. A comparative study of artificial bee colony algorithm. *Applied mathematics and computation*, *214*(1), pp.108-132.

Rylander, S.G.B., 2002. Optimal population size and the genetic algorithm. *Population*, *100*(400), p.900.

Chun-Feng, W., Kui, L. and Pei-Ping, S., 2014. Hybrid artificial bee colony algorithm and particle swarm search for global optimization. *Mathematical Problems in Engineering*, *2014*.

Eldrandaly, K.A., Hassan, M.M. and AbdelAziz, N.M., 2015. A modified artificial bee colony algorithm for solving least-cost path problem in raster GIS. *Applied Mathematics & Information Sciences*, *9*(1), p.147.

Mohamed, A.F., Elarini, M.M. and Othman, A.M., 2014. A new technique based on Artificial Bee Colony Algorithm for optimal sizing of stand-alone photovoltaic system. *Journal of advanced research*, *5*(3), pp.397-408.

Nozohour-leilabady, B. and Fazelabdolabadi, B., 2016. On the application of artificial bee colony (ABC) algorithm for optimization of well placements in fractured reservoirs; efficiency comparison with the particle swarm optimization (PSO) methodology. *Petroleum*, *2*(1), pp.79-89.

Hardiansyah, H., 2016. A Modified ABC Algorithm for Solving Non-Convex Dynamic Economic Dispatch Problems. *International Journal of Electrical and Computer Engineering (IJECE)*, *6*(6), pp.2621-2628.

Whittington, R., 2003. *Factors potentially limiting colony growth, foraging effort, and pollination efficiency of bumble bees in commercial tomato greenhouses* (Doctoral dissertation, Science: Biological Sciences Department).

Affenzeller, M. and Mayrhofer, R., 2002. Generic heuristics for combinatorial optimization problems. In *Proc. of the 9th International Conference on Operational Research* (Vol. 2002, pp. 83-92).

Harabor, D. and Botea, A., 2010, October. Breaking Path Symmetries on 4-Connected Grid Maps. In *AIIDE*.

Harabor, D.D. and Grastien, A., 2011, August. Online Graph Pruning for Pathfinding On Grid Maps. In *AAAI*.

Shah, H., Ghazali, R. and Hassim, Y.M.M., 2014. Honey bees inspired learning

algorithm: nature intelligence can predict natural disaster. In *Recent Advances on Soft Computing and Data Mining* (pp. 215-225). Springer, Cham.

Moersch, J. and Hamilton, H., Hybrid Vector Field Pathfinding.

Kim, D.H., Hai, N.T. and Joe, W.Y., 2017, December. A Guide to Selecting Path Planning Algorithm for Automated Guided Vehicle (AGV). In *International Conference on Advanced Engineering Theory and Applications* (pp. 587-596). Springer, Cham.

Dolinskaya, I.S., 2012. Optimal path finding in direction, location, and time dependent environments. *Naval Research Logistics (NRL)*, *59*(5), pp.325-339.

Khoury, D.S., Myerscough, M.R. and Barron, A.B., 2011. A quantitative model of honey bee colony population dynamics. *PloS one*, *6*(4), p.e18491.

Liang, J.J., Qu, B.Y., Suganthan, P.N. and Hernández-Díaz, A.G., 2013. Problem definitions and evaluation criteria for the CEC 2013 special session on real-parameter optimization. *Computational Intelligence Laboratory, Zhengzhou University, Zhengzhou, China and Nanyang Technological University, Singapore, Technical Report*, *201212*, pp.3-18.

Karaboga, D. and Basturk, B., 2007. A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. *Journal of global optimization*, *39*(3), pp.459-471.

Rangel, J. and Seeley, T.D., 2012. Colony fissioning in honey bees: size and significance of the swarm fraction. *Insectes sociaux*, *59*(4), pp.453-462.

Sharma, P. and Khurana, N., 2013. Study of optimal path finding techniques. *International Journal of Advancements in Technology*, *4*(2), pp.124-130.

Sonmez, M., 2011. Artificial Bee Colony algorithm for optimization of truss structures. *Applied Soft Computing*, *11*(2), pp.2406-2418.

Hadidi, A., Azad, S.K. and Azad, S.K., 2010, September. Structural optimization using artificial bee colony algorithm. In *2nd international conference on engineering optimization* (pp. 6-9).

Abdelaziz, A.Y., Ali, E.S. and Elazim, S.A., 2016. Flower pollination algorithm to solve combined economic and emission dispatch problems. *Engineering Science and Technology, an International Journal*, *19*(2), pp.980-990.

Chang, W.L., Zeng, D., Chen, R.C. and Guo, S., 2015. An artificial bee colony algorithm for data collection path planning in sparse wireless sensor

networks. *International Journal of Machine Learning and Cybernetics*, *6*(3), pp.375-383.

Jayalakshmi, B. and Singh, A., 2017. A hybrid artificial bee colony algorithm for the cooperative maximum covering location problem. *International Journal of Machine Learning and Cybernetics*, *8*(2), pp.691-697.

Liang, Y., Wan, Z. and Fang, D., 2017. An improved artificial bee colony algorithm for solving constrained optimization problems. *International Journal of Machine Learning and Cybernetics*, *8*(3), pp.739-754.

Garcia, R., Aycard, O., Vu, T.D. and Ahrholdt, M., 2008, December. High level sensor data fusion for automotive applications using occupancy grids. In *Control, Automation, Robotics and Vision, 2008. ICARCV 2008. 10th International Conference on* (pp. 530-535). IEEE.