THE DEVELOPMENT OF A ROBUST ALGORITHM FOR UAV PATH PLANNING IN 3D ENVIRONMENT

KOK KAI YIT

UNIVERSITI SAINS MALAYSIA

2016

THE DEVELOPMENT OF A ROBUST ALGORITHM FOR UAV PATH PLANNING IN 3D ENVIRONMENT

by

KOK KAI YIT

Thesis submitted in fulfilment of the requirements

for the degree of

Master of Science

March 2016

ACKNOWLEDGEMENTS

I would like to express my deepest gratitude to Dr. Parvathy for her guidance throughout the progress of this master research. I have learned from her that there is always more than one solution to a problem. Without her guidance, I would have never finished this thesis in this short time. I will forever be grateful for her financial assistance before I become fellowship holder at the University Science of Malaysia (USM).

I also wish to thank the staff of the School of Aerospace Engineering who willingly lent me a hand whenever I asked for help. I extend a special thank you to Kak Ayu for her kindness and patience in assisting me. I also thank the USM for the financial support and for providing me with an environment conducive to my research.

Finally, I would like to thank my friends and colleagues from the USM who made my life and my completion of my master's research truly memorable.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

UAV	Unmanned Aerial Vehicle
PSO	Particle Swarm Optimization
GA	Genetic Algorithm
DE	Differential Evolution
BBO	Biogeographic-based Optimization
IE	Infection Evolution
GPP	Global path planning
LPP	Local path planning
EA	Evolutionary algorithm
EC	Evolutionary Computation
VGA	Vibrational Genetic Algorithm
TOT	Time over target
CPPBBO	Chaotic Predator–prey Biogeographic-based Optimization
θ-QPSO	Phase Angle-encoded and Quantum-behaved Particle Swarm Optimization
DEQPSO	Hybrid Differential Evolution and Quantum-behaved Particle Swarm Optimization
DEPSO	Hybrid Differential Evolution and Particle Swarm Optimization
QPSO	Quantum-behaved Particle Swarm Optimization
μGA	Micro Genetic Algorithm
CPSO	Chaotic Particle Swarm Optimization
FEP	Fast Evolution Programming

UDE	Uniform-Differential Evolution
ODE	Oppositional-based Initialization Differential Evolution
QODE	Quasi-Oppositional-based Initialization Differential Evolution
SS	Smart Sampling
OGA/Q	Orthogonal Genetic Algorithm with Quantization
CHA	Opposition-based Chaotic GA/PSO Hybrid Algorithm
VPAC	Velocity-propelled average crossover
RNG	Random number generator
TNT	Tent map
LD	Low discrepancy
GLP	Good lattice point
OD	Orthogonal design
DMPSADE	Self-adaptive Differential Evolution with Discrete Mutation Control Parameters
Rank-JADE	Adaptive Differential Evolution with Optional External Archive and Ranking-based Mutation Operators
OvcPSO	Probabilistic Oppositional-based Particle Swarm Optimization with Velocity Clamping and Inertia Weight

LIST OF SYMBOLS

x_U, x_L, y_U, y_L	Upper and lower limit coordinate
θ	Angle
λ	Immigration rate
μ	Emigration rate
k	Number of species
n	Maximum amount of species in a habitat
I	Maximum immigration rate
E	Maximum emigration rate
x	Individual from the population
G	Current generation
NP	Population size
F	Differential weight
v	Velocity
c_1 & c_2	Acceleration coefficient
w	Inertial weight
r_g & r_p	Random value within [0,1]
g_{best}	Global best point
p_{best}	Local best point

PENGEMBANGAN ALGORITMA YANG MANTAP UNTUK MERANCANG LALUAN UAV DALAM PERSEKITARAN 3D

ABSTRAK

Penyelidikan menyeluruh telah dijalankan berkaitan perancangan laluan Pesawat Udara Tanpa Pemandu (UAV) dengan menggunakan algoritma evolusi seperti pengoptimuman kerumunan zarah (PSO), algoritma genetik (GA), evolusi kebezaan (DE), dan pengoptimuman berasaskan biogeografik (BBO). Bagaimanapun, prestasi kebanyakan algoritma ini akan menurun dari segi kos fungsi dan pengiraan apabila digunakan dalam sistem yang teguh. Oleh itu, algoritma baru yang dikenali sebagai evolusi jangkitan (IE) telah dibina dalam kajian ini. IE memudahkan pengiraan dan memaksimumkan kecekapan menjana perancangan laluan yang lebih baik dalam persekitaran 3D. 9 peta telah digunakan sebagai kajian kes, dan 100 simulasi telah dijalankan dalam setiap kes untuk mendapat purata prestasi algoritma. Semua simulasi telah dijalankan melalui MATLAB dengan pembayangan perancangan laluan UAV. Prestasi algoritma IE telah dibandingkan dengan PSO, GA, DE dan BBO pada tetapan optimum algoritma masing-masing. IE berjaya merancang laluan UAV yang lebih pendek dengan kadar kebarangkalian 92 peratus dalam 100 kajian kes. Selain itu, IE mencapai kelajuan pemprosesan yang lebih cepat berbanding dengan algoritma lain dengan kadar kebarangkalian 97 peratus. Oleh itu, algoritma IE menunjukkan potensi yang besar dalam perancangan laluan UAV.

THE DEVELOPMENT OF A ROBUST ALGORITHM FOR UAV PATH PLANNING IN 3D ENVIRONMENT

ABSTRACT

Significant research has been conducted on Unmanned Aerial Vehicle (UAV) path planning using evolutionary algorithms, such as Particle Swarm Optimization (PSO), Genetic Algorithm (GA), Differential Evolution (DE), and Biogeographic-Based Optimization (BBO). However, the performance of most of these algorithms tend to decline in terms of function and computational cost when dealing with robust systems. Thus, a new algorithm known as infection evolution (IE) was developed in this study. IE simplifies calculation and maximizes the efficiency of generating an improved path plan in a 3D environment. Nine terrain maps were used as case studies, and 100 simulations were carried out for each case to determine the average performance of the proposed algorithm. All simulations were performed using MATLAB with visualization of UAV path planning. The performance of the IE algorithm was compared with that of PSO, GA, DE, and BBO at their respective optimized settings. IE attained a 92% probability rate of achieving a short path length in 100 case studies. With regard to computational cost, IE attained a 97% probability rate of achieving a faster processing speed in comparison with tested algorithms. Therefore, the IE algorithm exhibits significant potential for UAV path planning optimization.

CHAPTER 1

INTRODUCTION

Section 1.1 briefly explains UAV path planning. Section 1.2 discusses the configuration of search space in path planning, and Section 1.3 elucidates the available algorithms for UAV path planning. Sections 1.4 and 1.5 present the problem statement and objectives of this research, respectively. Finally, Section 1.6 discusses the scope of the study, and Section 1.7 presents the outline of this thesis.

1.1 Outline on Path Planning

In the 21st century, researchers no longer prioritize flight speed and material development. Instead, researchers today consider the intelligent development of aircraft. Unmanned aerial vehicles (UAVs), in particular, have generated great interest among researchers for their potential in intelligent development. Such growing interest stems from the small size of UAVs [1] and their relatively lower cost compared with manned aircraft; these advantages make them appealing to the military sector seeking to reduce uncountable costs especially during dangerous missions [2, 3].

Many studies have been conducted on the development of UAVs because of the wide variety of their applications, including in the areas of surveillance [4, 5], traffic monitoring [6, 7], rescue missions [8, 9], aerial photography [10, 11], and fire-fighting [12]. Similar to manned aircraft, UAVs feature fixed wing and rotorcraft types. Given the small size of UAVs, multirotor UAVs such as the tricopter, quadcopter, hexacopter, and octocopter are developed to achieve excellent stability and manoeuvrability.

UAVs have been developed for various military and commercial purposes [13]. Researchers are developing new UAV technologies to reduce commands from pilots on flight missions, such as avoiding obstacles during flight without instructions from the pilot. The high manoeuvrability of UAVs, the reduced need for pilots, and the relatively low costs of such vehicles strongly motivate researchers to further improve UAV performance and ultimately develop UAVs with fully autonomous flight capabilities.

UAV path planning is the process of creating an optimum flight path from a starting point to final location. Hardware and software both play important roles in allowing UAVs to carry out path planning. Generally, hardware is used to receive signals from surroundings, process calculations, and act according to calculations, whereas software is used to analyze data from signals, trigger algorithms, and determine the next action of UAVs.

The common hardware used for UAV path planning, including microcontrollers, sensors, and motors, and the response performance of UAV path planning depend on hardware quality when using the same software. The common off-the-shelf microcontroller brands for UAV path planning are Arduino, Hobbyking, AutoQuad, and Crius with open source code. In terms of sensor selection, it is generally based on user requirements.

For example, an ultrasonic sensor can detect distances from obstacles, a thermopile can detect infrared, and a pair of thermopiles could maintain the flying level of a UAV by ensuring the same readings. Other available software for UAV path planning include the Mission Planner and UAV Planner whereby the algorithms

can be implemented into the software for path planning. Figure 1.1 shows the window of Mission Planner [14].

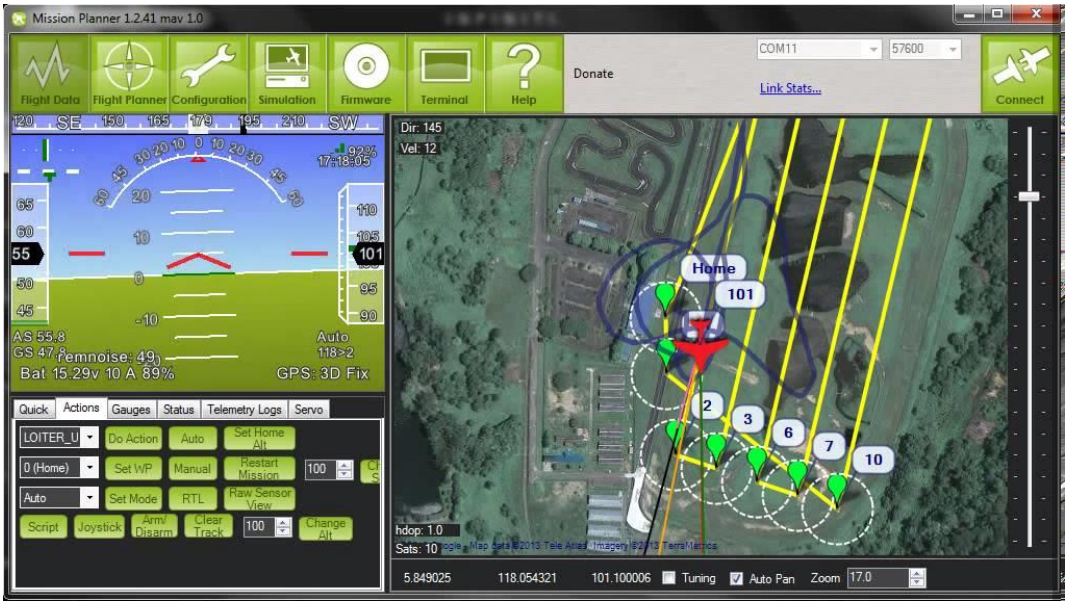

Figure 1.1: Mission Planner [14]

The two types of UAV path planning are global path planning (GPP) and local path planning (LPP). GPP generates a path using information of a certain area without the need for sensed information, whereas LPP is a continuous process of finding paths locally in real time for operation mission and vehicle safety [15]. In practice, GPP is initiated before a vehicle starts moving on the basis of previously acquired area information.

By contrast, LPP requires the continuous transmission of information from sensors to processors during movement from initial coordinates to final locations. Therefore, GPP usually occurs during the planning phase, and compared with LPP, GPP involves a larger scale of search as well as a longer duration. On the one hand, the large scale of search of GPP allows the generation of several efficient flight paths without being trapped. On the other hand, LPP should be accomplished at the

shortest possible time to avoid obstacles and maintain the stability of UAVs in real time, particularly because the response of UAVs is strongly affected by LPP. Hence, LPP can generate a safe flight path in a short period, but UAVs might be trapped before reaching their final locations.

The two types of representation for path planning are topological and metric [16]. Topological path planning uses identifiable objects or landmarks to generate a path. In UAV applications, the flight path produced from topological path planning comprises connections between identifiable intersections or landmarks; directions may include “fly over the bridge” and “turn right before the next corner.”

However, most research on the topological path planning of UAVs is performed in an indoor environment as it consists of more identifiable objects in comparison with an outdoor environment. Topological path planning usually uses voronoi diagrams or visibility graphs. Metric path planning applies the (x, y, z) coordinate system and is suitable either in indoor or outdoor environments. In metric path planning, direction commands include “fly to an altitude of 100 meters at 30 degrees for 200 meters.” This method works well in computer search algorithms. Figure 1.2 presents the difference between topological and metric path planning using the same map [16].

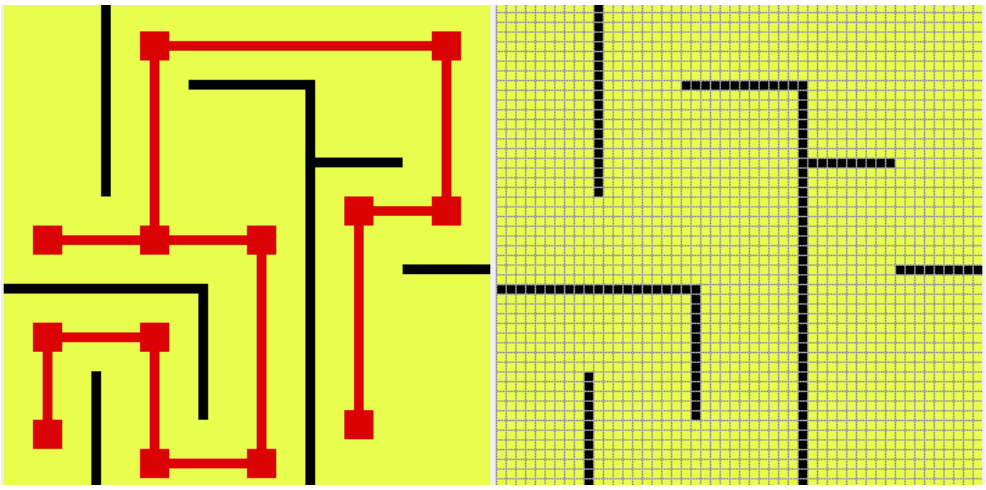

Figure 1.2: Topological (left) and Metric (right) Path Planning [16]

Path planning may be two-dimensional (2D) or three-dimensional (3D). 2D path planning only considers x- and y-axes when producing a path, whereas 3D path planning considers x-, y-, and z-axes, with the z-axis being the altitude range. Typically, non-flayable vehicles, such as cars and ships, use 2D path planning, whereas flyable vehicles, such as UAVs, use 3D path planning. Compared with 2D path planning, 3D path planning is more complicated and entails higher computational cost for the same problem because of its consideration of an additional axis.

1.2 Search Space Configuration

The three common search space configurations for path planning are cell decomposition, roadmap, and potential fields [16]. Cell decomposition configuration represents the world in grids, roadmap configuration forms connections between particular points, and potential field configuration resorts to mathematical fields to present the world. All of these space configurations can be used in either topological or metric path planning.

In cell decomposition configuration, the map is divided into grids and cells adjacent to other cells without overlapping. The method of traveling from one cell to an adjacent cell is known as connectivity graph. In fact, the function cost of cells in this method can be changed according to terrain information. Several types of cell decomposition include approximate decomposition, adaptive cell decomposition, and exact cell decomposition [16].

Approximate cell decomposition is the easiest to apply on a map as it allows the map to form regular grids with predefined sizes and shapes, as shown in Figure 1.3 [16].

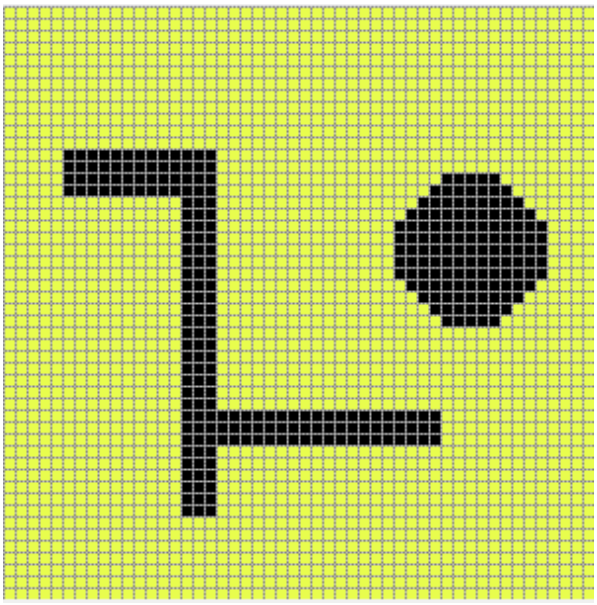

Figure 1.3: Approximate Cell Decomposition [16]

Adaptive cell decomposition reduces the number of cells in a map by using large cells in free space and small cells in the presence of objects. All cells are maintained in the same shape. Specifically, the number of cells is reduced as follows: the map is divided into four cells, and the cells with objects are continuously divided into four cells until all cells are completely empty or full, as shown in Figure 1.4 [16].

In exact cell decomposition, cells rely on the shapes and locations of obstacles in a map; thus, cells do not have predefined shapes and sizes. All cells connect to the edges of obstacles in the search space and thus allows the identification of a path, if any. Figure 1.5 shows an example of exact cell decomposition [16].

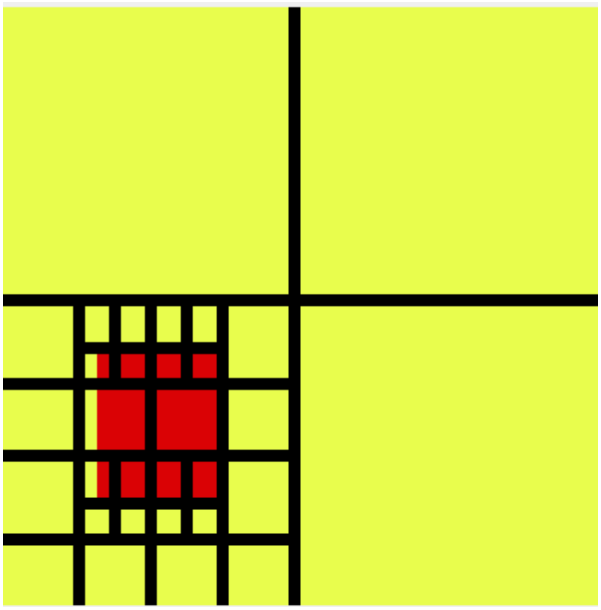

Figure 1.4: Adaptive Cell Decomposition [16]

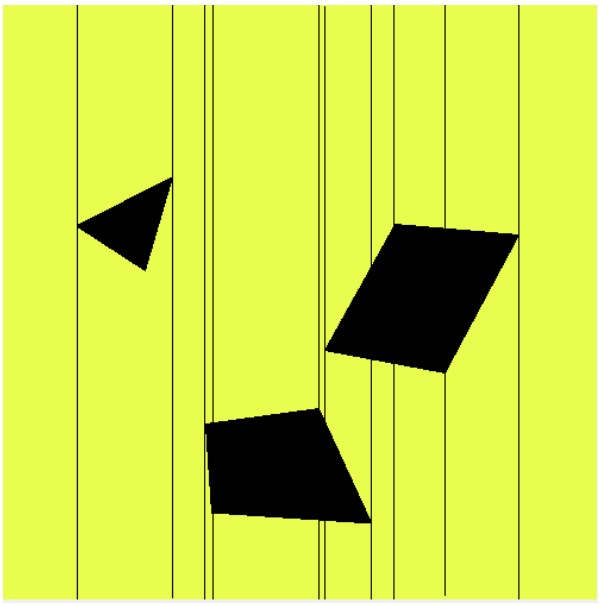

Figure 1.5: Exact Cell Decomposition [16]

Unlike cell decomposition configuration, roadmap configuration takes less time in searching for a path. Roadmap configuration only puts nodes at remarkable locations, such as building corners and landmarks. The number of nodes in roadmap configuration is smaller than the number of cells in cell decomposition configuration; hence, the former is easier to use to obtain a path from an initial point to a final point. However, the nodes in roadmap configuration should be remade when information is updated [16]. Visibility graphs, voronoi diagrams, and probabilistic roadmaps are examples of roadmap configuration [16].

Visibility graph requires a map with obstacles of a clearly defined polygon shape. Such requirement is due to straight lines form and become connected between the edges of polygonal obstacles. The paths from the initial to the final locations are generated by connecting these straight lines. However, some segments of the path may be too close to the boundaries of obstacles using a visibility graph. An example of a visibility graph is shown in Figure 1.6, in which the dotted line denotes the shortest path [16].

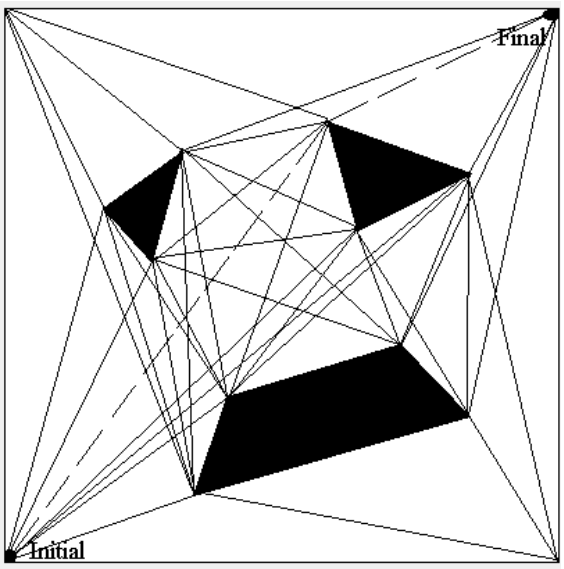

Figure 1.6: Visibility Graph [16]