ANALYSIS OF NATURE INSPIRED INTELLIGENCE IN THE DOMAIN OF PATH PLANNING AND SEARCHING WITH CONSIDERATION OF VARIOUS PARAMETERS

A

Thesis

Submitted to

For the award of

DOCTOR OF PHILOSOPHY (Ph.D.)

in

COMPUTER SCIENCE AND ENGINEERING

By

Monica Sood

(41400097)

Supervised By

Dr. Sahil Verma

Co-Supervised By

Dr. Vinod Kumar Panchal

LOVELY FACULTY OF TECHNOLOGY AND SCIENCES
LOVELY PROFESSIONAL UNIVERSITY
PUNJAB
2019

CANDIDATE'S DECLARATION

I hereby declare that the thesis entitled "Analysis of Nature Inspired Intelligence in the Domain of Path Planning and Searching with Consideration of Various Parameters" submitted by me for the Degree of Doctor of Philosophy in Computer Science and Engineering is the result of my original and independent research work carried out under the guidance of Supervisor Dr. Sahil Verma, Associate Professor, School of Computer Science and Engineering, Lovely Professional University, Punjab, and Co-Supervisor Dr. Vinod Kumar Panchal, Former Director & Scientist-G DTRL Lab, DRDO, New Delhi. This work has not been submitted for the award of any degree, diploma, associateship, fellowship of any University or Institution.

Monica Sood
(Registration Number: 41400097)
School of Computer Science and Engineering,
Lovely Professional University,
Phagwara, Punjab-144411,
India
Date:

CERTIFICATE

This is to certify that the thesis entitled " Analysis of Nature Inspired Intelligence in the Domain of Path Planning and Searching with Consideration of Various Parameters " submitted by Monica Sood for the award of the degree of Doctor of Philosophy in Computer Science and Engineering, Lovely Professional University, is entirely based on the work carried out by her under my supervision and guidance. The work reported, embodies the original work of the candidate and has not been submitted to any other university or institution for the award of any degree or diploma, according to the best of my knowledge.

Supervisor	Co-Supervisor
Dr. Sahil Verma	Dr. Vinod Kumar Panchal
Associate Professor,	Former Director & Scientist-G,
School of Computer Science and Engineering,	DTRL Lab,
Lovely Professional University,	DRDO
Phagwara, Punjab-144411,	New Delhi
India	India
Date:	Date:

ABSTRACT

Nature being a source of inspiration always fascinates the humans to develop novel concepts that can solve any NP hard problem. Although the advancements in the field of computing and technology have shifted the focus of human towards latest techniques, the main source of inspiration behind these techniques is always the nature and its natural phenomenon. This research work also focuses on the nature inspired concepts to determine the solution of the optimal path planning problem. In this research work, path planning problem is handled by proposing a hybrid concept of CS-FAPP (Hybrid Cuckoo Search with Firefly Algorithm for Path Planning) algorithm. The overall research objectives of thesis includes the path planning using proposed hybrid CS-FAPP (Hybrid Cuckoo Search with Firefly Algorithm for Path Planning) algorithm, evaluation of algorithm by alteration the parameters, and comparison with hybrid CS-BAPP (Hybrid Cuckoo Search with Bat Algorithm for Path Planning) algorithm.

The planning of optimal path is an important research domain due to its vast applications in the fields of robotics, simulation & modeling, computer graphics, virtual reality estimation & animation, and bioinformatics. In optimal path planning, it is important to determine the collision free shortest and optimal path. There may be various aspects to determine the optimal path based on different types of available obstacles during the path and different types of workspace environment. This research work aims to identify the optimum path from defined source to destination for the unknown workspace environment consists of static obstacles. Among swarm intelligence concepts, Firefly Algorithm (FA), Bat Algorithm (BA), and Cuckoo Search (CS) algorithm are considered due to adaptability of these concepts according to problem definition. For this experimentation, swarm intelligence based hybrid concept of CS-FAPP is proposed to find the optimal path from a defined source to destination. The algorithm of CS-FAPP has been experimented, analyzed, and compared with CS-BAPP algorithm along with other individual algorithms as per the defined research objectives. Although work collaboration and intelligence behavior of

swarm agents are efficient to solve NP hard problems but the hybridization of concepts make the solution of problem more efficient. Another reason for the hybridization is that individual BA and FA can lead to problem of trapping between local optima. This obligates to hybridize the individual concepts of BA and FA with some more efficient algorithm like CS having clever behavior and brood parasitic property to store their egg in other bird's nest.

In the proposed CS-FAPP approach, CS algorithm handles the obstacles in the path and FA determines the optimized path. The considered workspace region is assumed as collection of binary pixels values: 0 and 1 where value 1 indicates obstacle free white pixel and value 0 is black pixel with obstacle. In the proposed hybrid algorithm, the obstacles on the workspace are handled with cuckoo search based property with an assumption of considering obstacle as worst nest for cuckoo egg and path planning is performed by other algorithm of FA.

The algorithms are initially tested on the benchmarks functions and applied to the path planning problem. The considered benchmark functions are Rosenbrock Function, Michalewicz Function, Ackley Function, Easom Function, De Jong Function, Schwefel Function, Rastrigin Function, Griewank Function, and Shubert Function. After achieving the successful performance outcomes using proposed CS-FAPP approach as compared to other concepts, it is considered for the experimentation on the application of path planning.

For path planning, Google based real time satellite images of different regions of India are considered. The selected regions consist of terrain features of urbanization, barren, water, hilly (rocky), and vegetation with major focus on vegetation regions. All the images possesses different size and taken from different regions representing the terrain features with diversity. The satellite image 1 of size 267*265 pixels consists of mainly the vegetation region along with barren and urbanization terrains and captured from the nearby region of Mahodra, Rajasthan, India. The satellite image 2 of size 307*240 pixels also contains the terrain features of vegetation, urbanization and barren. This image is taken from the Kelwara region of Rajasthan, India. The satellite image 3 of size 203*258 is taken from the Tulera Region, Alwar

(Rajasthan), India. This image consists of mainly the terrain feature of dense vegetation along with some urbanization sector. The next satellite image 4 of size 331*248 is taken from the border line regions of Rajasthan and Madhya Pradesh states of India. The nearby regions are Ghatta region of Rajasthan and Parwah region of Madhya Pradesh, India. This image consists of terrain features of water, vegetation, barren, and urbanization. The last satellite image 5 of size 405*316 is taken from the nearby Hedri, Maharashtra, India which consists of mainly the terrain regions of hilly and vegetation.

The results of path planning are evaluated in terms of simulation time, minimum iterations required to obtain optimal path length, success rate, and error rate. The results are evaluated in the iteration manner and overall results. The iteration-wise results are determined for the iteration difference of 10 for all the satellite images using considered and proposed algorithms. The best path length and best simulation time refers to the optimized path length obtained after each 10 iterations and time taken to obtain that best path length after each 10 iteration interval respectively. The optimal path length achieved for all the satellite images is same for the proposed CS-FAPP, hybrid algorithm of CS-BAPP, and the other individual algorithms. There is the difference of the simulation time and iteration number at which the optimal length achieved for the different algorithms. The evaluated results indicate the obtained optimal path lengths of 320 pixels, 261 pixels, 246 pixels, 318 pixels, and 376 pixels for the satellite images 1-5 respectively. The proposed CS-FAPP approach has achieved these optimized path lengths with minimum number of iterations of 36, 23, 27, 31, and 26 respectively. The simulation time is also optimum time as compared to all the other concepts. The evaluated results of proposed hybrid CS-FAPP algorithm are compared with the individual FA, BA, & CS along with the comparison of hybrid CS-BAPP concept based on the mentioned parameters.

In all the aspects of benchmark functional testing and evaluation metrics (simulation time, minimum iterations required to obtain optimal path length, success rate, & error rate) testing, proposed hybrid CS-FAPP algorithm outperforms in comparison with another proposed hybrid concept of CS-BAPP algorithm and individual BA, FA, and CS.

ACKNOWLEDGEMENT

I am extremely grateful to my guides, Dr. Sahil Verma and Dr. Vinod Kumar Panchal, who have been a continuous source of knowledge, inspiration, motivation and encouragement during the entire course of this research work. Despite their busy schedules, they have been available at every step, devoting time and energy and the much needed counsel and advice. This enabled me to sail through the tough times and complete this enormous task.

A special thanks to the management of Lovely Professional University for supporting me in the best possible manner and facilitating me in balancing my work and my research. The doctoral programme of LPU has made it possible for me to pursue my dream of research and upgrade my knowledge.

I thank my friends, especially Ms. Sapna Saini, Ms. Shabnam Sharma – a good friend, for being there for me and motivating me at every step to achieve this goal. Without their support and patience during the numerous ups and downs, their constant encouragement - 'you can do it' and 'you are almost there', completion of this journey would have been impossible. The support extended by them throughout the course of my research is beyond words and my literary capabilities.

My sincere feeling of gratefulness also goes to my parents and brother who always motivated me in all the endeavours of my life including this research work in LPU. I am also thankful to my husband Mr. Varun Luthra who has provided the full support to me during the entire period of my research work. My special thanks to my daughter Kashvi for giving me joyful and happy moments during the entire journey of my research work. Finally I would like to thank each and every person who has directly and indirectly helped and motivated me in this herculean task.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

SI	Swarm Intelligence
NIC	Nature Inspired Computing
FA	Firefly Algorithm
CS	Cuckoo Search
BA	Bat Algorithm
ABC	Ant Colony Optimization
ABC	Artificial Bee Colony Optimization
PSO	Particle Swarm Optimization
ACO	Ant Colony Optimization
SDS	Stochastic Diffusion Search
AFS	Artificial Fish Swarm Algorithm
FPA	Flower Pollination Algorithm
BBO	Biogeography Based Optimization
IWD	Intelligent Water Drops Algorithm
CS-FAPP	Hybrid Cuckoo Search and Firefly Algorithm for Path Planning
CS-BAPP	Hybrid Cuckoo Search and Bat Algorithm for Path Planning
GWSO	Glow-Worm Swarm Optimization
GSO	Gravitational Search Optimization
GA	Genetic Algorithm
SA	Simulated Annealing

NN	Neural Network
BPF	Bacterial Potential Field Algorithm
TSP	Travelling Salesman Problem
MIRAN	Modified Indicative Routes and Navigation
MOVNS	Multi-Objective Variable Neighbourhood Search
PRP	Probability Road Planner
PPP	Probability Path Planner
MEJ	Method of Evolving Junctions
VLSI	Very Large Scale Integration
HSI	Habitat Suitability Index
RPP	Robotic Path Planning
PPSO	Polar Coordination PSO
RNA	Robot Navigation Ant algorithm
MASC	Multi Scout Ant Corporation
RRT	Rapid-Exploring Random Tree
UCAV	Uninhabited Combat Air Vehicle
SLEPSO	Switching Local Evolutionary PSO
UAV	Unmanned Aerial Vehicle
D^2PSO	Dynamic Distributed PSO
FL	Fuzzy Logic
TS	Tabu Search
SA	Simulated Annealing
FLC	Fuzzy Logic Controller

ICA	Imperialist Competitive Algorithm
POI	Points Of Interest
BINN	Bio-Inspired Neural Network
GACRPP	Genetic Algorithm-based Compliant Robot Path Planning
FSVM	Fuzzy Support Vector Machine
GRNN	General Regression Neural Network
SACO-MH	Simple Ant Colony Optimization Meta-Heuristic Algorithm
EP	Evolutionary Programming
DV	Differential Perturbed Velocity
GUI	Graphical User Interface

CHAPTER 1
INTRODUCTION

This chapter illustrates the nature inspired computing, categorization of nature inspired computing techniques based on artificial immune systems, geo-sciences, modelization of human mind, and swarm intelligence. Also, the basic concept of optimal path planning, applications of path planning in various domains, and different types of path planning techniques are discussed in this chapter. The chapter finishes up with the research objectives, contributions, and thesis organization.

1.1. INTRODUCTION

Nature being a source of inspiration has fascinated the humans to develop efficient algorithms that can solve even the NP hard problems as well [1]. Nature inspired computing consisting of various innovative techniques that are developed by imitating the behavior of nature [2]. Nature inspired computing algorithms have major features of self-organization, adaptive, emergent, distributed, and autonomous. There are four major categories of nature inspired computation based on artificial immune systems, geo-sciences, modelization of human mind, and swarm intelligence [3]. This thesis focuses on swam intelligence concepts for the application of optimal path planning. The Concept of optimal path planning can be defined as a way to plan a shortest & efficient path for the defined source to destination within the efficient time interval [4].

Swarm intelligence (SI) techniques are derived from the collective intelligence attributes of social species [5] [6]. There are various SI techniques derived from the attributes and working of different social species such as Artificial Bee Colony Optimization (ABC) [7], Bat Algorithm (BA) [8], Ant Colony Optimization (ACO) [9], Flower Pollination Algorithm (FPA) [10], Cuckoo Search (CS) [11], Intelligent Water Drops Algorithm (IWD) [12], Artificial Fish Swarm Algorithm (AFS) [13], Firefly Algorithm (FA) [14], Stochastic Diffusion Search (DFS) [15], Biogeography Based Optimization (BBO) [16], and Particle Swarm Optimization (PSO) [17].

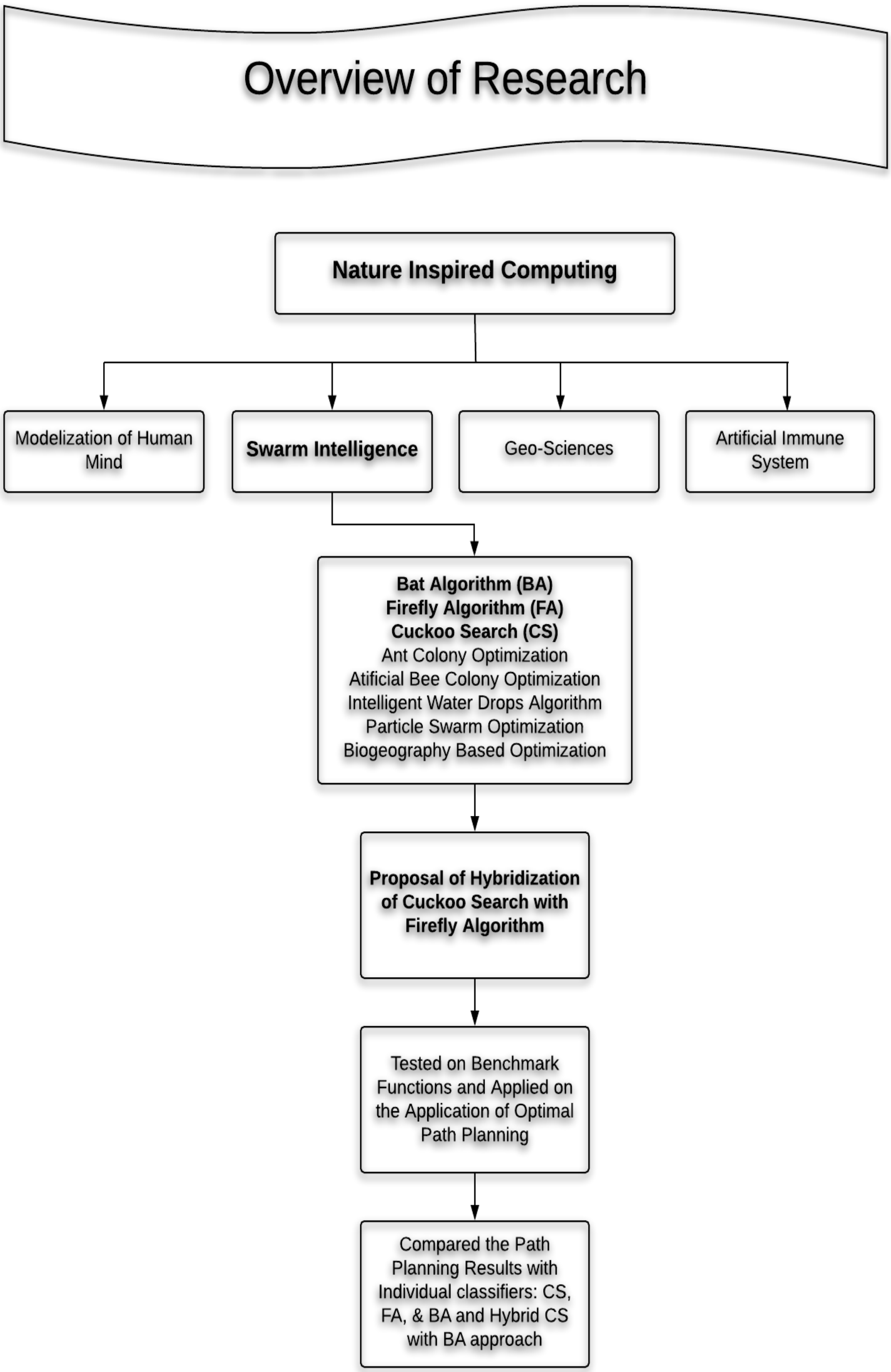


Figure 1.1: Thesis Research Work Overview

This thesis work proposes a hybrid concept of Cuckoo Search with Firefly algorithm for path planning (CS-FAPP). Initially, the efficacy of proposed CS-FAPP system is tested with the benchmark functions and then applied on Google based satellite images of different regions situated in India for path planning. The evaluated results are compared with the hybrid CS with BA for path planning (CS-BAPP) and individual algorithm of CS, FA, & BA. The overview work flow of the proposed concepts for optimal path planning using nature inspired computing has been illustrated by figure 1.1.

1.2. NATURE INSPIRED COMPUTING

Nature has always played an indispensable role in the lives of living organisms. It has always contributed in the form of water, land, air, food, shelter etc. in order to maintain the existence of living beings [18]. The list of contribution is not only limited to food or water, nature has given different phenomenon in the fields of physics, chemistry, medicine etc. Several processes and mechanisms are explained and are even inspired from nature and natural effects. Various laws like law of thermodynamics, law of action and reaction, law of gravity or the revolution of earth around the sun are few concepts which are explained or even originate from nature.

With passage of time and the day by day advancements in technology and science, human brain has also grown. Numerous machines are invented and the process of invention is boosting. Humans are considering nature as their source of inspiration to solve computationally rigorous and complex problems.

Nature being a supreme inspirational source motivates humans to invent new techniques to solve any of the complex problems. Its scope is not limited to any of the field. The terminology which is used to address techniques, procedures, process or any other situation for which nature is considered as a source of inspiration to get results is "Nature Inspired Computing" [19][20]. Researchers around the globe are functioning on the concepts of Nature Inspired Computing. It is not bounded to a particular field. The concept is used in almost every area whether it's engineering, biology, medicine, physics or any other [21].

The concept of Nature Inspired Computing is categorized into four main conceptual categories [3], which are described as below:

1.2.1. Swarm Intelligence
1.2.2. Modelization of Human mind
1.2.3. Artificial Immune System
1.2.4. Geo-Sciences

This categorization serves as a basic building block in order to solve different decision making problems, optimization problems, decidability problems and many more. Different categories have different methods or different problem solving techniques. The categories are discussed as follows:

1.2.1. Swarm Intelligence

From the ancient years, natural social species fascinated the individual and researchers for the seamless intelligence work solutions. These social species can be small ants, termites, wasp, bee, flying birds, school of fishes and other animal groups also [22]. These species work in collaborative behavior. Even each individual agent have their own agenda but the work seems to be in a well organized manner and their coordinated behavior can achieve complex goals. The daily task of social insects include food search, efficient brood feeding, spreading alarm, respond to external challenges, extending/building nests etc. All the agents of a particular colony species interact with each other directly or indirectly and solve the problem in a robust and optimized manner. In swarm intelligence, behavior analysis of social species, modeling their behavior and use the modeled form to develop an artificial swarm concept is included. So, swarm intelligence can be defined in the following manner [23].

Swarm Intelligence is an artificial concept inspired from the collaborative multiagent social species that work intellectually to design intelligent system and optimized algorithm [24].

The researcher 'Beni' have initially used the term "swarm" during cellular robotic system in which agents organizes them with nearest neighbour interaction [25] [26].

After that it is being used for the various optimization area like science, industry etc. There is a great importance of swarm intelligence techniques to find the optimized solutions of problems for industrial research & science projects. The examples of optimization problems cover the computational biology, network design in telecommunication, shape optimization, time tabling and train scheduling etc. For the test case of any novel swarm concept, authors usually consider Travelling Salesman problem (TSP) [27]. The motive of TSP problems is to reach each city once with minimal value of travelling distance, where number of vertices or cities is predefined. Another example of optimization problem can be Protein Folding which is a common problem for the fields of physics, biochemistry, computational biology, and molecular biology.

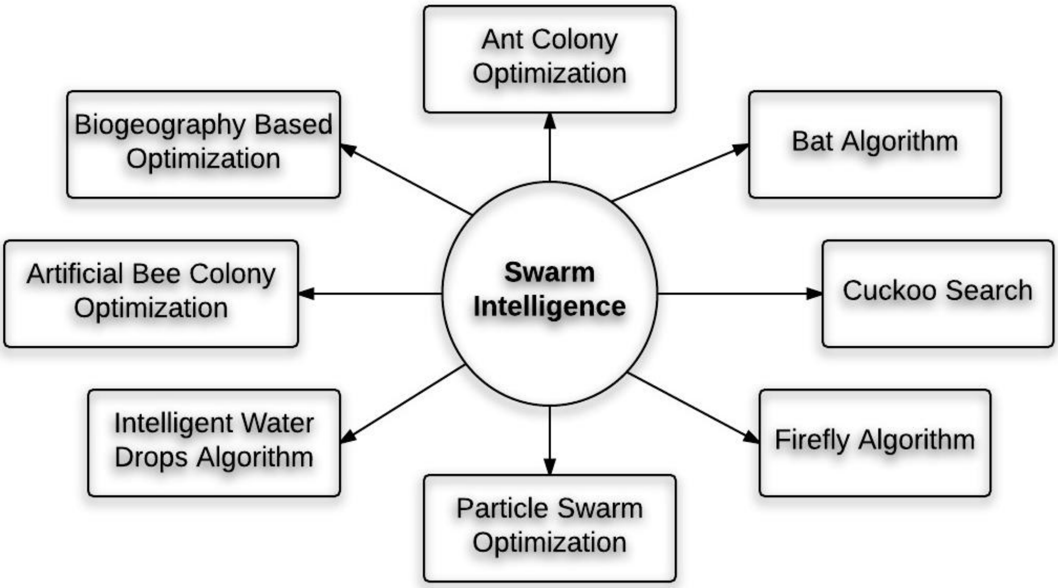


Figure 1.2: Various Techniques of Swarm Intelligence

1.2.1.1. Self Organization in Social Species

The global cooperation of these social species to work collaboratively is due to strategy of self organization. Self organization is a dynamic strategy where agents create global system by interaction of local level components [28]. All the lower level agents interact purely based on local information and maintains the global system strategy without any external ordering influence. There are four basic ingredients for self organizations. These factors are multiple interactions, amplification of fluctuations, negative feedback and positive feedback.

1.2.1.2. Swarm Intelligence Techniques

Based on the variety of social species of insects and animals, there exists a number of Swarm Intelligence techniques [29]. The popular techniques based on SI are listed (shown in figure 1.2) Bat Algorithm [30], Cuckoo Search [31], Biogeography Based Optimization [32], Ant Colony Optimization [33], Firefly Algorithm [34], Artificial Bee Colony Optimization [35], Intelligent Water Drops Algorithm [36], and Particle Swarm Optimization [37]. From the above techniques, concepts of BA, FA, and CS have been exploited to determine the solution for optimized path planning problem.

1.2.2. Modelization of Human Mind

In this technique, as the name suggests, the role played by human mind is vital. The capabilities of human mind are used in order to find out or to optimize the solution. The technique of Modelization of Human mind differs from other techniques due to the involvement of the various features or advanced concepts based on the capability of human mind. This technique includes concepts which are not approachable to the other categorized techniques in Nature Inspired Computing. The concepts like, imprecision or uncertainty in order to achieve solutions with high degree of robustness, optimization and cost effectiveness, are computed effectively using this technique.

To determine the efficient resolution of the computational model problem, the absolute illustration of the problem both mathematically as well as logically is necessary. Further, different computer models are required to verify the properties and simulation features. The techniques in Modelization of Human Mind are generalized to obtain an enhanced appreciative knowledge of the logical or mathematical as well as behavioral aspects. The techniques based on this concept are Wisdom Technology [38], Anticipatory Computing [39], Fuzzy Set Theory [40][41], Granular Computing [42], and Rough Set Theory [43][44]. These techniques are shown in figure 1.3.

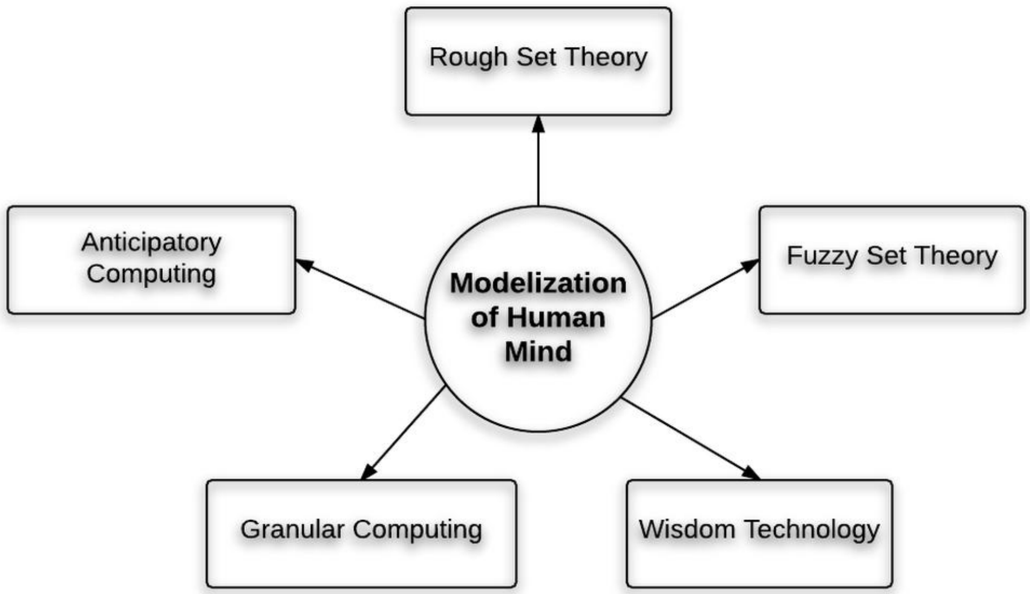


Figure 1.3: Various Techniques of Modelization of Human Mind

The concept of Rough Set Theory deals with the problems with imprecise, vague or incomplete data. In this modelling technique, approximate sets are considered to extract some useful information. Approximate sets are sets with very few available evidences required for their description. Another concept of Fuzzy Set Theory is imitated from the concept of partial existence and reasoning. It consists of many functionalities and operators like membership functions, union, intersection, t-norms, co-norms etc. to name as few. The overall functionality of the system depends only on the truth value of its membership function which is further calculated with the help of fuzzy set operations.

Wisdom Technology considers real life constraints or situations in order to make satisfactory decisions. The Wisdom modelling technique is used in both of the artificial as well as natural systems. The model of Anticipatory computing deals with the concepts for future decision making. In this model predictions or assumptions are considered in order to make decision about the future. Anticipatory computing forecasts about the future events by characterizing and then, further reducing the cause of uncertainty. Thus, providing accurate and timely decisions. Granular computing also known as Information Granularity involves the process of granulation, in which problem statement is divided into small granules. Granulation involves the process of construction, which follows bottom-up approach to convert the granules at lower level to the high-level granules and the other process is decomposition, which is basically reverse of construction [45]. It follows top-down approach to convert granules at higher level to the low level granules.

1.2.3. Artificial Immune System

It is a theoretical approach with the aim of problem solving and acquiring optimized results [46]. This system is inspired from the theoretical immunity system and contains certain immunity system related factors to solve the problem and find the optimized solution effectively. This system has its application in research domains of biology, robotics etc. The immunity system of our body protects us from the pathogen caused infection. The two major immune systems, Innate and Adaptive immune system protects our cells from the diversifying molecules, cells or organs. The computational categorization of the Artificial Immune System is discussed with the techniques of Evolutionary Computing [47], Neuro Computing [48], Membrane Computing [49], Cellular Automata [50] and DNA Computing [51]. These techniques are shown in figure 1.4.

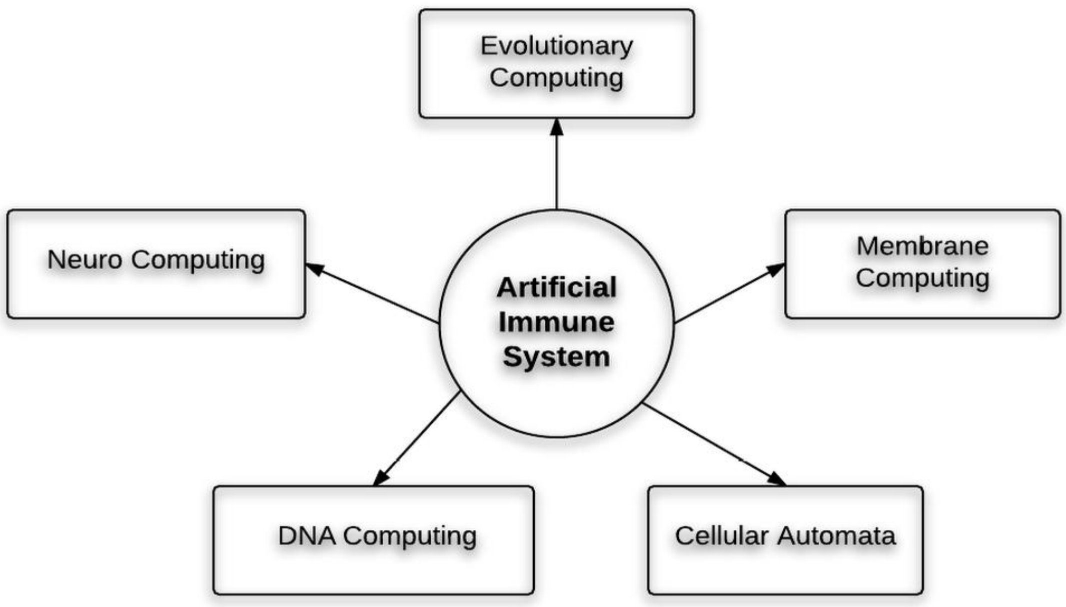


Figure 1.4: Techniques based on Artificial Immune System

Evolutionary Computing is the most accepted computing technique in the field of biology. This technique aims to find out the diversity in different living organisms by considering numerous features like the similarities and the differences among them. It is based on the concepts of Mutation, for which a fitness function is considered to select the corresponding mutation and recombination. In Neuro Computing technique, a learning rule is their according to which the neural architecture will be trained and information is processed. Once the network learning is done, it can be trained according to the particular task to get the output results. It is the property of neural network that it can learn from the available circumstances i.e. it has the ability to learn from the given example or even to generalize things.

Membrane Computing can be illustrated as procedure of performing computations with the help of living cell's membrane. 'P system' is the most basic model for membrane structure. Membrane structure is basically a model which the hierarchical arrangement of the membranes. Skin membrane or plasma is the outermost membrane and region that can be illustrated as the space attained by any membrane. This membrane computing concepts is similar to the concept of parallel computing but in non-deterministic manner. It has its application in remote sensing and is used to determine the solution for the NP hard and NP complete problems.

Cellular Automata is discrete modulation of finite number of grid of cells having finite number of grid dimensions as well as finite number of states of cells. The technique of Cellular Automata has its applications in many of the areas like in physics, mathematics, biology, complexity science etc. Whereas, DNA Computing is widely applicable in biology related areas. The concept of DNA Computing consists of the concepts of computations, bio-technology and mathematics. In the Artificial form of DNA Computing only the basic concept of DNA Computing without any existence of molecules is considered. But in the Wet form of DNA Computing, actual lab experiments are done with the consideration of actual DVA molecules.

1.2.4. Geo Sciences

The concept of Geo Sciences consists of all of the scientific techniques related to the planet Earth. Our planet consists of land, water, air, living beings etc. and on the basis of this categorization is done named as biosphere, atmosphere, hydrosphere etc. Under the earth's crust energy is transformed from one form to another. The geological phenomena like the volcano eruption, earthquakes, mountain range etc. shows the transformation of the energy under the earth crust. The concept of Geo Sciences can be categorised into the following techniques of Big Bang Crunching [52], Volcanic Eruptions [53], Plate Tectonics [54], Ocean Currents [55], Tidal Waves [56]. These techniques are shown in figure 1.5.

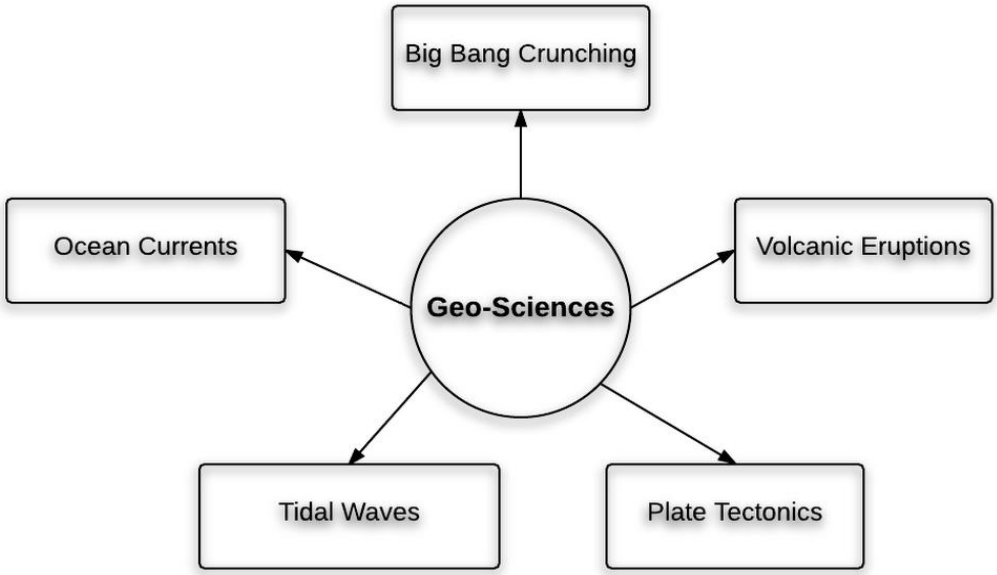


Figure 1.5: Techniques based on Geo-Sciences

Big Bang Crunching is the concept which explains the birth of the universe. The theory of big bang illustrates that how from a tiny state which was dense in nature, universe is transformed to its current state. It is a conceptual model which only describes what has happened. The theory of big bang focus on the expansion or stretching of the universe and how different energies were transformed. Volcanic Eruption is discharging of the lava, molten rock or gas from a volcanic outlet. The volcano is said to be in active state when the lava is erupting from the outlet of volcano. The tendency of explosion depends on the amount of the trapped gas in magma/lava and the flow of the lava. The appearance of volcano can be different depending on the features. The different types of volcanoes are: caldera volcano, composite volcano and shield volcano.

According to the concept of Plate Tactonics, earth's outer layer is divided into numerous plates which may be set of small as well as large plates. The movement of both of the types of plates is relative to each other. The plates when either slip or converge or diverge each other, activity like earthquake, volcanic eruptions takes place. Most of the part of the Earth is covered with oceans. The concept of Ocean Current is the movement of the water of oceans which may be continuous or directed in nature. Different forces like the rotation of the earth, the temperature variations or wind etc. are responsible for the generation of the ocean currents. In order to determine the climate, the concept of ocean currents holds great significance. Tidal waves are generated when the Sun, earth and moon interacts. The gravitational interactions between them results in waves of little depth known as tidal waves.

In this research work, swarm intelligence based concepts are employed for the application of shortest path planning. The contemplation of swarm intelligence techniques for this application is due to properties of swarm intelligence concepts such as collective cleverness & intelligence, self-organizing behavior, decentralized, and scattered throughout surroundings. Further, the basic of path planning concepts, categorization, and applications are discussed.

1.3. PATH PLANNING

Path planning can be illustrated as the primary motion planning problem in the community of robotics [57]. Motion planning involves an autonomous navigation system with an ability to determine collision-free path and move without any human assistance. In general, motion planning consists of two dissimilar but supportive tasks of path planning and trajectory planning [58]. Path planning incorporates an autonomous system capable of determining collision-free path from specified source to destination. Whereas, trajectory system includes the motion of the system from specified source to destination by following the path evaluated through path planning.

1.3.1. Basic Path Planning

The problem of path planning is an intricate task which has been a motivating factor for researchers from several research domains including: simulation, very large scale integration system, geographic information system, computer graphics, animations and gaming [59]. A typical formulation of path planning problem could be considered as *"the piano-movers problem"* [60], where it undertakes the crash free path determination by moving a piece of furniture say sofa from one room to other in a cluttered house. The simple version of path planning is to identify the shortest and safe path-route from the defined source spot to defined destination spot in given workspace with obstacles [61]. If W is the given workspace with obstacles *ob* and defined source & destination are *S* & *D* respectively, then figure 1.6 specifies the path planning problem.

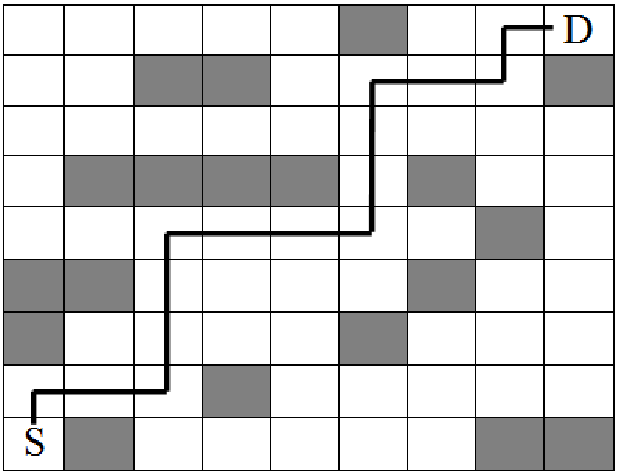


Figure 1.6: Simple Path Planning Problem

Let *Sys* denotes the system and W defines the workspace region with obstacles *Ob* such that $Ob \subset W$. Here, obstacles are un-determined objects with any size or shape and can be present anywhere in the workspace region. In 2-dimension, this workspace region is linearly bounded. The system is allowed to proceed through the whole workspace region except the regions with obstacle. In basic path planning, the start and destination points in workspace region are predefined with obstacles information. The objective is to compute the shortest or optimized direction from beginning point to the termination point without any impact even with the availability of obstacles. The figure 1.7 demonstrates the concept of basic path planning in 2-dimensional workspace with the consideration of an example. The task of path planning in terms of algorithm with input and output can be explained as:

- Inputs: The inputs to the algorithm are the starting and destination points along with the defined obstacles in the workspace region.

- Output: This indicates the gradual system movement i.e. how the system will move in the workspace region in order to visit the destination-point by handling the obstacles in the pathway.

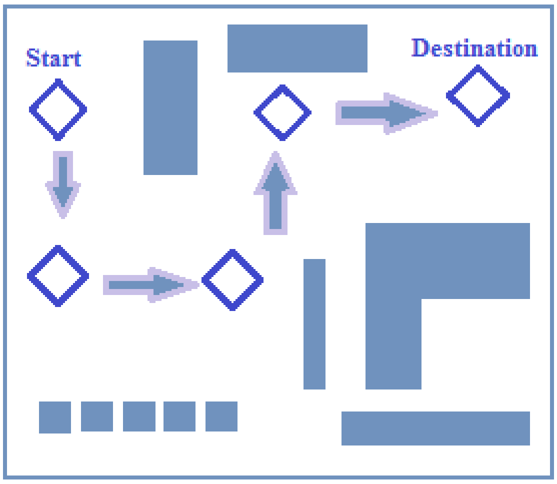


Figure 1.7: Example of Path Planning in 2-Dimensional Workspace

1.3.2. Path Planning Classification

There are several categories to classify the path planning problem. It can be broadly classified as static and dynamic path planning. The two categories are further classified into known and unknown environment information as presented in figure 1.8. A static environment consists of motion-less obstacles i.e. the environment does not contain any moving obstacles. Whereas, dynamic path planning as the name suggests contains moving obstacles. The environment information could be known or unknown.

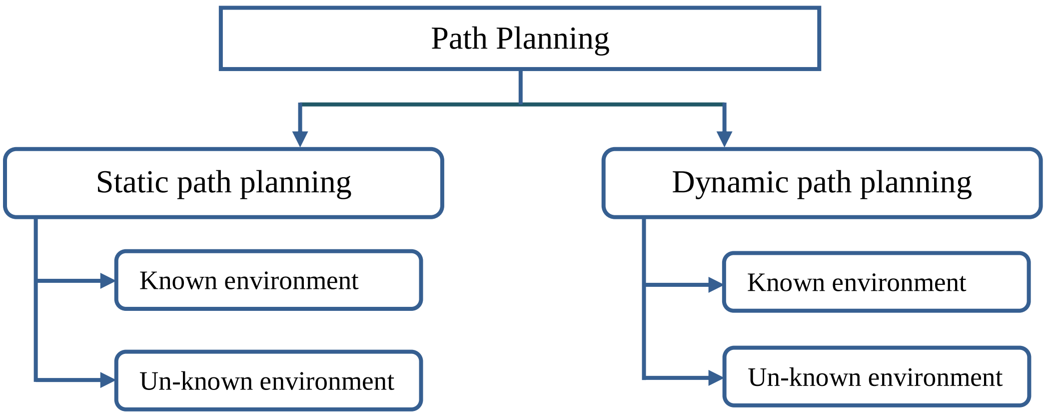


Figure 1.8: Path Planning Classification

The static path planning in which there is the preceding knowledge of the obstacles and environment is categorized as static path planning in a known environment. In case, there is no information or little information available of the environment in the presence of motion-less obstacles then it is called as static path planning in an un-known environment. There is the work of various researchers on the static path planning with the consideration of different algorithms. In a static environment, an A* heuristic based path planning technique is presented by [62] [63] [64] [65]. Dolgov et al. [62] demonstrated the path planning concept for automated vehicles in an unknown environment. The output of the proposed A* technique is then optimized by numeric non-linear optimization technique. In an unknown environment, Sudhakara and Ganpathy [63] also implemented an obstacle free path planning approach for robots. Guruji et al. [64] designed an autonomous system for robotic path planning which calculates the value only before the collision detection. Silva et al. [65] modifies the cost function in A* algorithm to identify the route for mobile robot. In virtually static environment [66] [67] [68] [69] the concept of path planning has been discussed to control the moving characters from source to destination. A corridor map method is developed [66] and implemented [67] by planning paths inside the corridors for different character types. Jaklin et al. [68] have introduced a novel technique of path planning called as Modified Indicative Routes and Navigation (MIRAN) to plan paths for characters in virtual environment. Ganpathy et al. [69] enlarged the simulation workspace area by modifying the existing ACO algorithm for the path planning of mobile robots. The results are evaluated by performing implementations at real time and simulating environment with the available information related to environment to avoid static obstacles. Mobile robots are very popular to plan the path from one point to other. Hidalgo et al. [70] developed a novel method named a Multi-Objective Variable Neighbourhood Search (MOVNS). The mobile robot is made an understanding about the workspace in order to detect safe and smooth paths with minimum path length. Lee and Kim [71] used directed acyclic graph along with genetic algorithm to implement the robotic path planning. The authors considered the static and controlled grid workspace and convert into set of nodes. Then a directed acyclic graph is generated and based on the graph multiple paths are generated from the initial-point to final-point. The proposed method is evaluated on 20 map scenarios with different conventional methods using three evaluation measures. Santa et al. [72] considered skeletonization algorithm from image processing techniques to plan the route for mobile robots in a static and controlled environmental conditions. Initially, the image considered is passed through number of filters to make it obstacle free. Then, the skelentonization concept is applied to determine the shortest path within two defined points. The resulting solution is the cost efficient global optimal path for mobile robots.

Path planning in dynamic environments as well is the most attentive research fields over past few years. If there is preceding information about the environment and the obstacles present are moving obstacles, then it is categorized in the category of dynamic path planning within known environment. If there is no information regarding environmental conditions, then it is classified in the category of dynamic path planning in un-known environment. Dynamic environments are more complex and lead to more changes over time as obstacles are free to move. The probability path planner or probability road planner (PRM) [73] is an effective technique to plant paths in a motion/dynamic environment. PRM has its applications in many of the areas solving the problem of path planning. The paths detected through this technique are long and depend on the local properties of the workspace environment. Brož and Apu [74] developed a path planning system technique in partially known or unknown/dynamic environmental conditions for mobile robots. This technique has been presented as the amalgamation of grid and graph based approaches. The graph is used to determine reachable points from all nodes and grid represents the discrete structure of the environmental constraints. The experimentation results are evaluated on different datasets in terms of cluster amount, allocated memory, adaptation and path finding time. Another hybridization of techniques has been discussed by Montiel et al. [75] in the environmental conditions which contains static or dynamic obstacles for mobile robots path planning. This proposed technique is combination of bacterial evolutionary algorithm and artificial potential field, thereby named as bacterial potential field algorithm (BPF). The efficiency of this presented approach is showed by obtaining statistical test values of five different problems. Li et al. [76] also proposed a novel approach in the dynamic environmental conditions with dynamic obstacle types for the system based path planning. The technique is named as method of evolving junctions (MEJ) which is a combination of differential equations and intermittent diffusion. The proposed numerical algorithm is efficient in term of finding a global optimum path with low cost values. Further, Nimoniya et al. [77] presented the hybrid approach to ensure a constraint or obstacle aware effective navigation in dynamic environment representations. Path planning for vehicles in dynamic environment is an important aspect to consider. For instance, the traffic accident case which become extremely important for the emergency vehicle like ambulance to reach on time for the rescue operation. Zhao et al. [78] implemented two stage path planning model for emergency vehicles in dynamic environment. Initially time travel predicting functions are developed based on the real life road traffic. Then shuffled frog leaping algorithm is applied to obtain the k number of paths in the first stage and grouping of the paths in the second stage is done to find the one optimum path. The experimentation results obtained when applied on the floating vehicle data of Beijing shows the efficiency of the proposed algorithm. A bio-inspired neural network path planning technique is proposed by Jianjun et al. [79] for automated underwater vehicles. The autonomous vehicle plans the path source to destination without any prior knowledge except the information about the target location. The larger obstacles are removed automatically and computing efficiency is improved with the help of target attractor. A recent development in automated vehicles for dynamic path planning is done by Hu et al. [80]. Initially central line is constructed and then the s-p coordinate system generates the path candidates. A novel cost function is considered to identify the static and dynamic safety and comfortability of the optimal path. The proposed method is evaluated by designing test scenarios for single/multi lane roads with both the moving and static obstacle types.

In another scenario, the concept of path planning can also be categorized into local or global approach [81] [82]. Global path planning includes the identification of the complete and collision-free unbroken path through the system before any movement. Global path planning is also called off-line path planning. On the other hand, local or on-line path planning satisfies several constraints including time, path etc. to identify step by step position to reach the destination point [83] [84]. Global path planning results in a low-resolution path and is not appropriate in case of dynamic or unknown environment. The another concept of local path planning results into high- resolution path but it is not appropriate in case when the destination point is far away.

Another classification of path planning problem is proposed on the basis of implementation techniques: classical techniques and meta-heuristic techniques. The path planning problem has been developed with extensive research over the decades. The problem was distinctly defined during 1970's. The next two decades experience the development of some ideal path planning problem solutions but those were not practical in nature. That was the time when the classical techniques were majorly implemented to propose the solution for path planning problem. The techniques mainly include cell decomposition method, potential field and sampling based methods. In cell decomposition method, the workspace is divided into number of simple cells and the relationship between the adjacent cells is computed. The goal is to identify a collision free path from the defined source and destination. Initially, the cells containing the specified source and destination points are identified. The source and goal cells are connected by bridging them with a series of connected cells. The decomposition of cells is based on the dependency of the object. Figure 1.9 presents the solution sequence of cells from source *S* to goal *G* in an object dependent cell decomposition path planning problem. The application of cell decomposition method in path planning can be observed by the work of Kuan et al. [85] and Noborio et al. [86].

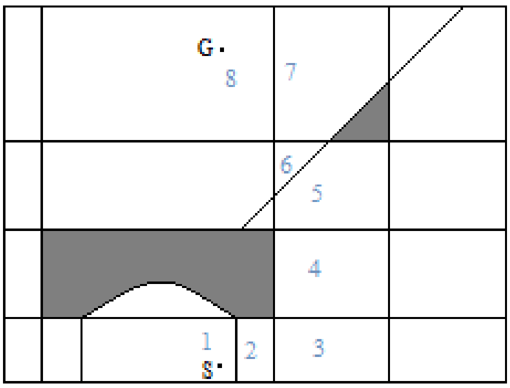


Figure 1.9: Path Planning by Objet Dependent Cell Decomposition Method

Further, Potential field method constructs potential field values based on the obstacles and destination point. The potential field value is low at destination point and high at obstacles. It could also be considered as repulsive and attractive forces which are assigned to obstacles and destination respectively. Barraquand et al. [87] constructed numerical potential field values to identify the obstacle free path from specified source to destination. The concept of potential field was implemented by Khatib and Mampey (1978) [88] and Hogan (1984) [89] to identify the optimal path from specified source to destination. Also, a review and construction of potential field can be seen in the work of Koditschek [90] to simplify the path planning problem.

Hou et al. [91] developed an algorithm to identify optimal path from source to destination by combing the approaches of cell decomposition and potential field. The workspace environment information was known in advance and several obstacle scenarios were considered to calculate the efficacy of proposed algorithm.

Another classical method for solving path planning problem is sampling based. Sampling based methods are popular than the other classical methods due to their ability to identify optimal path in real world and complex scenarios. Sampling based path planning approach includes methods like probabilistic road map and rapid-exploring random tress [92]. The two approaches construct graphs of the randomly sampled connecting points. However, probabilistic road map method has gained more popularity than the rapid-exploring random tress due to its efficient applicability in high-dimensional workspace. In a given workspace, the probabilistic road map is initially constructed and then is stored in graphical format. The graph nodes indicate the obstacle free optimal path [93]. Visibility graph and Voronoi diagram are trendy road map techniques to determine the shortest collision free route. The application of probabilistic road map method can be found from the work done by Kavraki and Latombe [94] and Boor et al. [95].

In classical methods, in case of large workspace area, the time complexity is higher and usually gets trapped in problem of local optimum. Due to this reason, meta-heuristic techniques gained their popularity in solving path planning problems. Meta-heuristic techniques include several algorithms like neural network (NN), particle swarm optimization (PSO), Fuzzy logic, genetic algorithm (GA), simulated annealing (SA), tabu search, and ant colony optimization (ACO), to name a few. Mohanty and Parhi reviewed the classical and heuristic techniques for path planning [96]. In their review, the authors concluded that the research in path planning was dominated by classical methods before the year 2000 and after that heuristic techniques gained their popularity. Heuristic methods are problem dependent optimization techniques which were introduced to solve problems more swiftly. The techniques are determined to find an approximate problem solution in case when an exact solution is not found. Meta-heuristic techniques are high level heuristic techniques with more powerful problem solving mechanisms. Meta-heuristic techniques employ heuristic techniques in case of large workspace and utilize the capabilities of heuristic techniques for better solution results. They lead towards the solution results of multi-objective and multi-level problems. Nowadays, meta-heuristic concepts are broadly accepted for the path planning problems in comparison with classical methods. Among meta-heuristic concepts, swarm intelligence concepts have gained a more acceptance of concepts due to their problem solving behaviour with optimized solutions. Therefore, this thesis work concentrates on swarm intelligence techniques for path planning.

Table 1.1: Path Planning Application Areas

Application	**Description**
Robotics	Path planning is a vital task for both industrial and service robots. In industrial applications, robots work in a predictable environment whereas service robots operate in an unstructured and unpredictable environment.
Simulation and Modelling	Virtual prototyping in designs, dynamics and kinematics leads to achieve realistic virtualization [97].
Computer Graphics and Animation	The integration of animation and graphics with artificial intelligence lead to the high-level usage of path planning techniques to achieve real time animation [98].
Biology	It is an unexpected application area of path planning. In chemistry & biology, the path planning techniques are used to identify protein folding gateways [99] and ligand docking [100].
Surveillance	For planning and surveillance of indoor environments, path planning techniques are used to identify the paths covering the entire workspace also called as “Watchman route problem” [101].

1.3.3. Path Planning Applications

The planning of optimal path is an important research domain due to vast applications of optimal path planning in the robotics, very large scale integration (VLSI), computer graphics, simulation & modeling, virtual reality estimation & animation, and bioinformatics. Table 1.1 lists some interesting application areas of path planning.

This thesis research focuses on the optimal path planning application to find best possible shortest route from defined initial point to final destination-point by handing the path obstacles. Here, optimal path is determined using swarm intelligence based FA, CS, and BA. The proposed hybrid CS-FAPP algorithm is defined using CS and FA. Another comparable algorithm CS-BAPP also makes use of CS along with BA. Further, the basic of these selected swarm intelligence algorithms and path planning using these algorithms is discussed.

1.4. BASIC FUNDAMENTALS OF CS, FA, AND BA

We have selected swarm intelligence concepts of FA, CS, and BA for path planning. The superior properties of these algorithms urge to consider these concepts. The basic fundamentals of these algorithms are discussed here.

1.4.1. Cuckoo Search

Cuckoo search is a swarm intelligence based optimization algorithm, proposed by scrutinizing the behavioural aspects of some species of cuckoo bird [102-104]. Cuckoo birds are very captivating not only because of their sweet sound they make but also due to their attribute of brood parasitic to lay their eggs into the host bird’s nest. Cuckoo species like *guira* and *ani* lay their eggs in host bird’s nests and remove the eggs of host birds to increase the hatching probability of their own eggs. If the host bird recognizes the cuckoo bird’s egg, it will either throw the egg or will forsake its own nest and build another nest elsewhere. Some female parasitic cuckoos like *tapera* have evolved a special ability to mimic the pattern and colour of eggs of some selected host species. Also, the timing of laying eggs and the selection of host bird’s nest by some cuckoo bird species is pretty amazing. The hatch process of cuckoo bird from eggs is earlier than the host eggs which makes cuckoo bird more confident to lay the egg in host nest.

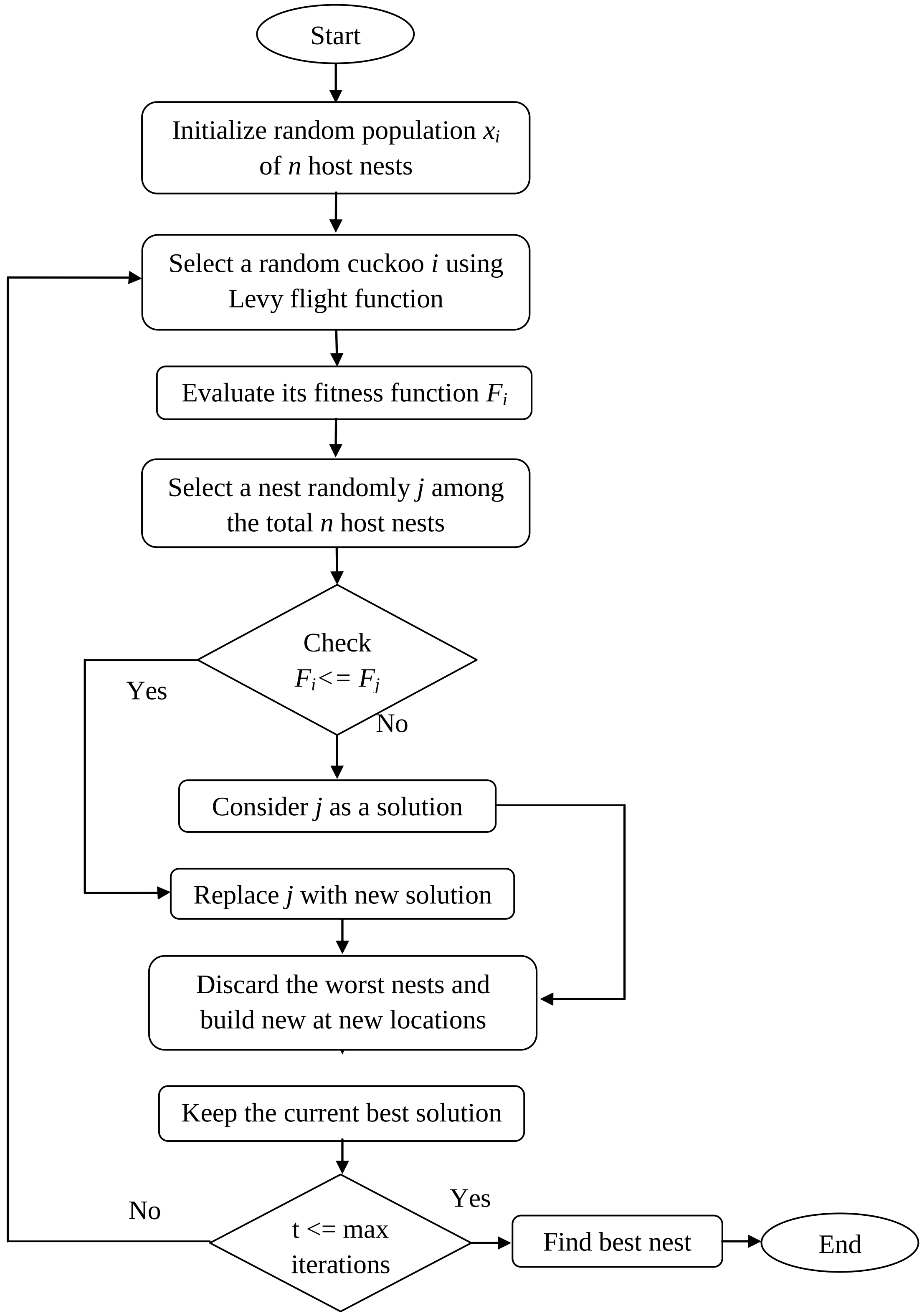


Figure 1.10: Workflow of Cuckoo Search Algorithm

At each iteration, there is the use of levy flight attribute by cuckoo birds for the generation of nests. In cuckoo search algorithm, levy flight is an arbitrary walk specified by series of jumps selected from a power law tailed probability density function. In cuckoo search, the process of levy flight is performed by randomly selecting the position of egg. The values of objective function is calculated and compared. The selected egg is moved to new coordinates if the values are better than the previous values. Thus, the process offers an optimal random search pattern. The scale is controlled by the multiplying levy flight with step size α. The flowchart of cuckoo search algorithm is illustrated by figure 1.10.

Cuckoo search algorithm is simplified by three rules specified by Yang and Deb as listed below:

1. There can be only one egg laid at once & dumped at arbitrarily selected nest by cuckoo bird.

2. Next generation will be carried over by the best nests with high quality eggs.

3. The probability of detection of cuckoo's egg by host bird is $p_a \in [0,1]$ and the numbers of host nests available are also fixed.

In case of rule number 3, the host bird will either throw the cuckoo bird's egg or will forsake its own nest and build another nest elsewhere. It can be approximated by probability p_a of n nests replaced new nests.

1.4.2. Bat Algorithm

Alike to the CS algorithms, Bat Algorithm (BA) is also introduced by Yang in 2010 [105]. The authors developed the BA by considering the working behaviour of micro-bats. Micro-bats are only mammal species that have wings and works on the basis of echolocation. There are different species of micro-bats with various sizes from small bee size to giant bats. Micro-bats use the advanced property of echolocation to detect any prey, hurdle during the pathway, and to detect the prey in night darkness. The flowchart of bat algorithm is illustrated by figure 1.11. The working of BA can be structured as per the defined regulations.

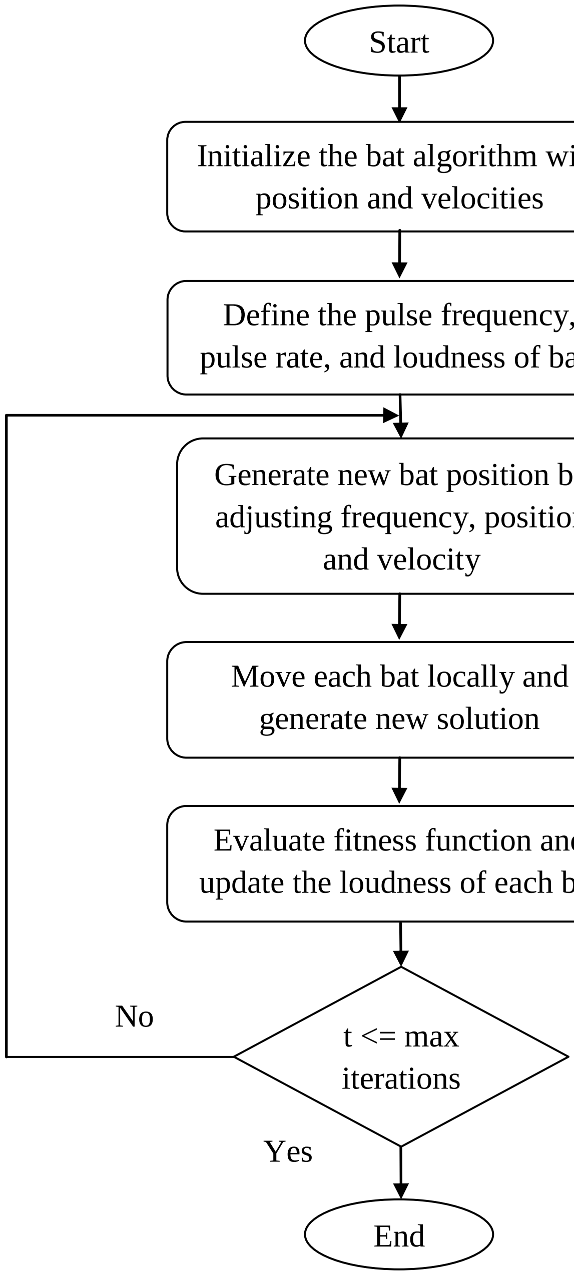


Figure 1.11: Workflow of Bat Algorithm

1. Bats have some special magical property to differentiate the hurdles in the pathway and prey/food. Bats use the sonar (echo-location) property to observe the distance from the target.

2. The initial parameters of loudness A_0, varying wavelength λ, frequency f_{min}, position x_t, and velocity v_t, are used by bats to determine the initial target. As

per target prey or hurdle, bats can change the wavelength and pulse emission rate r, where $r \in [0, 1]$.

3. There is the possibility of number of means that bats can own to change the loudness but it is assumed to consider higher loudness value A_0 to lower loudness value of A_{min} as the standardized values.

1.4.3. Firefly Algorithm

The meta-heuristic firefly algorithm was also introduced by Xin-She Yang in 2007 [106-107]. The algorithm is inspired by the captivating flashing property of fireflies. The fireflies use their bioluminescent property to characterize other fireflies and for mating as well. Besides signalling for mating, the flashing light also serves to warn the predators. Thus the primary purpose of the flashing light is to communicate and to attract other fireflies for mating. Firefly algorithm randomly generates an initial population of feasible solutions. The firefly population works in collaboration with an aim of sharing the knowledge with each other in order to ensure the best solution result in the search space. Each firefly works in a multidimensional search space and dynamically updating attractiveness based on firefly and its neighbour's knowledge. The flowchart of firefly algorithm is illustrated by figure 1.12. The algorithm is approximated with the help of following three rules or constraints based on the features of fireflies:

1. All fireflies are unisex i.e. no gender specificity will be there. One firefly will be attracted towards other brighter firefly regardless of its sex.

2. The attractiveness of a firefly is proportional to its brightness i.e. less brighter firefly will be attracted and move towards the brighter firefly. The brightness decreases with an increase in distance and if there is no brighter firefly then, the particular firefly will move randomly.

3. The value of objective function determines the brightness or attractiveness of a particular firefly.

Firefly algorithm has two key aspects which are light intensity and attractiveness. One firefly will be attracted towards the other firefly which is brighter than itself. Yang states that in firefly algorithm light intensity (L.I) decreases with an increase in square of the distance (d^2). With an increase in distance, light becomes weaker as light absorption increases. The value of fitness function (ff) for each solution value is calculated and it is proportional to the light intensity of firefly with solution s i.e. $L.I(s) \propto ff(s)$.

Attractiveness: The attractiveness β of a firefly is proportional to its brightness and it decreases with increase in distance d. Thus, attractiveness of a firefly is proportional to light intensity $L.I$ and it can be calculated with the help of Equation (1.1).

$$\beta = \beta_0 \, e^{-\gamma d^2}$$

… Equation (1.1)

Where γ is the light absorption coefficient and β_0 is the attractiveness at $d = 0$.

Motion of firefly: If s_i and s_j are two fireflies then the Euclidean distance between the fireflies can be evaluated by using Equation (1.2).

$$d_{ij} = \|s_i - s_j\| = \sqrt{\sum_{k=1}^{n} (s_{ik} - s_{jk})^2}$$

… Equation (1.2)

The motion of one firefly attracted towards the other firefly depends on its current position, the attractiveness and random walk consisting of the randomization variable α as mentioned in the Equation (1.3).

$$s_i = s_i + \beta_0 \, e^{-\gamma d^2} * \left(s_i - s_j\right) + \alpha \left(rand - \frac{1}{2}\right)$$

… Equation (1.3)

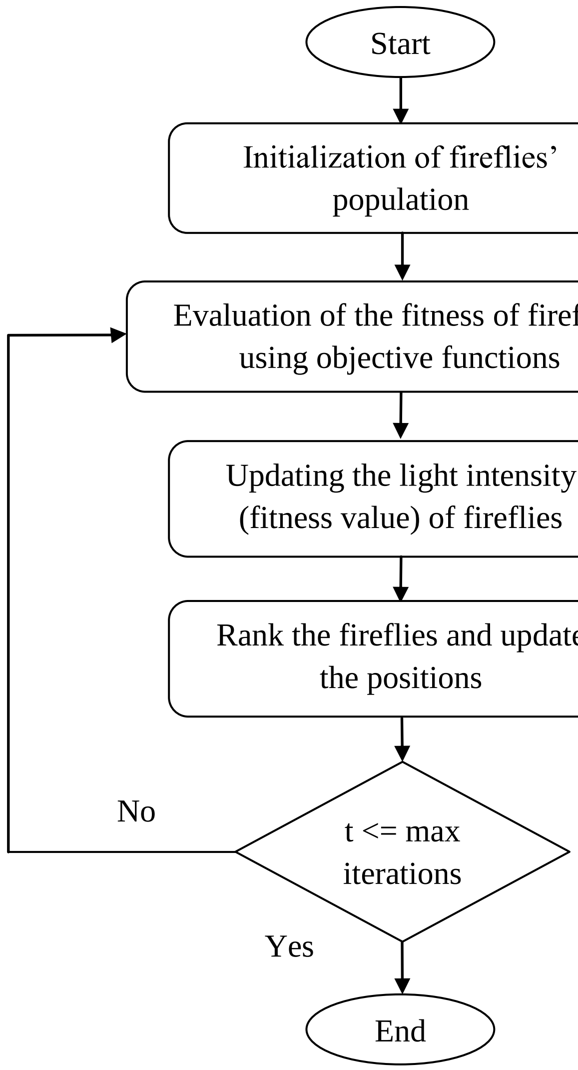


Figure 1.12: Workflow of Firefly Algorithm

1.5. PATH PLANNING USING INDIVIDUAL CS, FA, AND BA

Path planning using individual concepts is discussed as there is need to define the hybrid concepts using these individual concepts. Path planning using individual algorithms is also presented based on the experimentation on satellite images of different regions of India (Refer to Chapter 4 for Datasets of satellite images). The main goal to use the individual swarm intelligence techniques of CS, BA, and FA is to determine the results with these individual algorithms. The considered CS, BA, and FA use their own individual properties to handle the present obstacles and to find the optimal path from source point to destination point in a static and unknown environment. The behavioural property of fireflies of getting attracted towards brighter firefly is considered to identify optimal path in defined workspace. The obstacles are detected based on the light intensity value. The workspace is considered as set of values varying according to the light intensity value at particular point. On the other hand, CS algorithm works with an assumption of considering obstacle as worst nest for cuckoo egg to handle the obstacles. BA uses the frequency based echolocation system to detect the obstacles and handle those obstacles. Here, the workspace region is considered as collection of binary pixels values: 0 and 1 where value 1 indicates obstacle free white pixel and value 0 is black pixel with obstacle. The algorithm is similar for all the individual CS, BA, and FA with their own find finding and obstacle handing properties which are discussed above. The workflow of the algorithm is discussed in figure 1.13. The step by step algorithm is discussed here:

Algorithm

Step 1: Consider the input Google based satellite image and define the source s_p and destination d_p for the path planning.

Step 2: Apply Morphological operation to reduce the blocked paths and unnecessary area gaps between the source to destination.

Step 3: Initialize the parameters of swarm intelligence algorithms for n-agents possessing random position x_i in d-dimensional search space.

Step 4: Evaluate the size g_{best_i} and position x_{best_i} for the minimum fitness function value.

Step 5: Handle the obstacles based on the different properties of different swarm based algorithms.

Step 6: When the obstacle is detected then the Euclidian distance of the neighbour pixels from destination point is calculated to find the next point for the movement of swarm agents. If p_i and dp_j are pixel point and destination point respectively then the Euclidean distance between them can be evaluated by using equation (1.4).

$$d_{ij} = \|p_i - dp_j\| = \sqrt{\sum_{k=1}^{n}(p_{ik} - dp_{jk})^2}$$

… Equation (1.4)

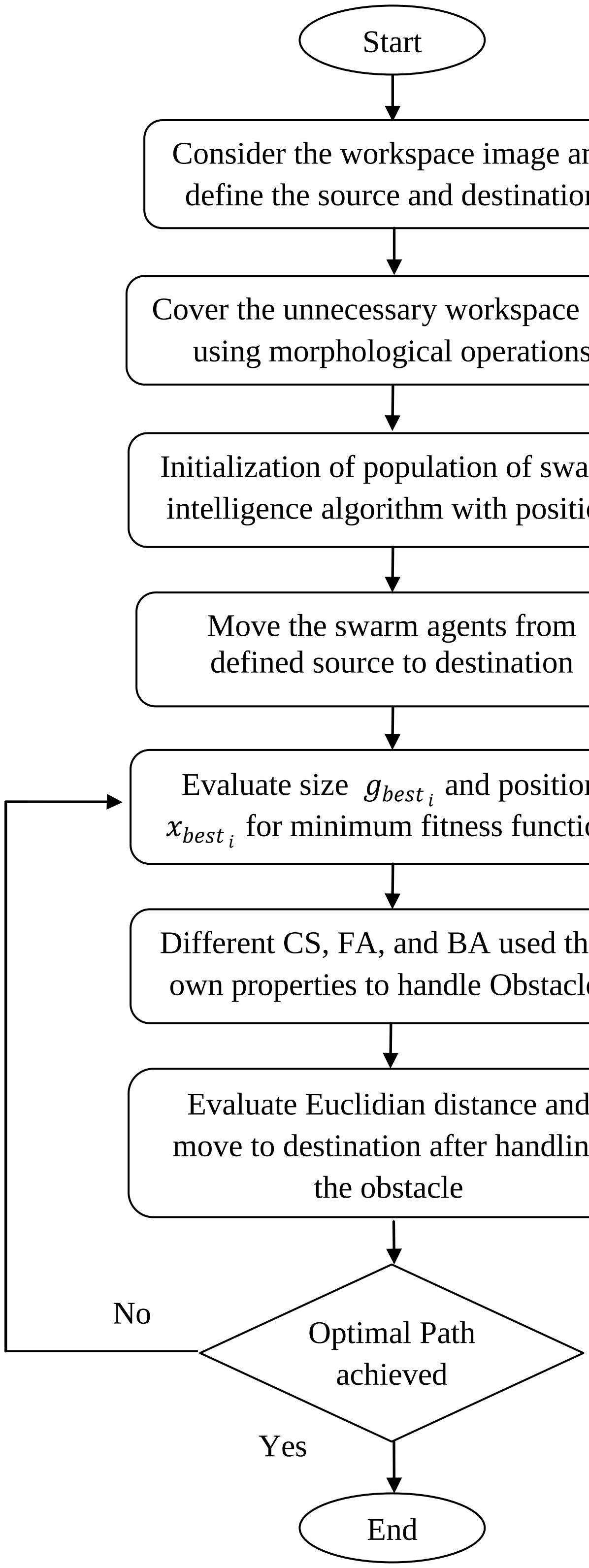


Figure 1.13: Workflow of Path Planning using Individual Algorithm

After, handling the obstacle, swarm concepts again proceed towards the destination by calculating the Euclidean distance from the neighboring pixels to destination point.

Step 7: Evaluate the swarm intelligence based global optimum solution for all the pixel points in the way from source to destination.

Step 8: Obtain the final shortest optimum path from source to destination.

The final optimal path obtained using different swarm intelligence concepts is same but the simulation time and minimum number of iterations required to obtain that path differs. The obtained path length is also same as all the algorithms are able to obtain the optimal path. The evaluated results with individual algorithms are discussed in Chapter 4 along with the comparison results of hybrid CS-BAPP and proposed hybrid CS-FAPP.

1.6. PROBLEM STATEMENT AND MOTIVATION

The problem of finding and planning the optimum path in different environments with obstacles has always allured the researchers from multiple disciplines. The increasing applications and usability of path planning in real world over the past few years is the key source of attraction for researcher. The application areas of path planning involve industrial applications, military applications, airports and many more. These increasing applications are one of the motivation factors to consider optimal path planning concept.

Although, there are plenty of existing methods on path planning (discussed in Chapter 2) but there is always a scope to improve the results based on its basic parameters of completeness, path length, success rate and computational time. Moreover, Google Map results can also be further enhanced for the betterment of path planning results in dense vegetation and unidentified areas. As per the terms and conditions of Google Map, there may be a chance of actual conditions differing from the map results [108]. It means results on Google map which are considered to be most accurate can also be improved as Google map routing depends on the actual conditions, assumption of risk, etc.

Another motivating factor of path planning is more availability of research publications on optimum path planning but comparatively lesser implementation in real life applications. In this research work, we have applied this path planning concept on real time Google map satellite images of various regions of India. The design and implementation of the automated system for path planning to work in real world environment is the prime objective of this work. This research work focuses on different terrain features of selected Indian regions to present its applicability in real world scenarios. Real world scenario is different than the virtual prototypes. They are more complex, unstructured and large in nature. The obstacles present in the path can have any shape, size and location as the considered environment is unknown for the method. Also at the same time, it is expected to find the optimum path by achieving efficiency in time consumption and complexity.

Moreover, there are lot of existing algorithms and applications (discussed in Chapter 2) for the path routing that works in urban area mainly in the known environment. There are other algorithms also that work in unknown environment but the performance and success rate of those path routing concepts is less. In these considered Google based satellite images, we have focused on the urban area along with other terrain features with major interest on vegetation area for the optimum path planning with unknown area and static obstacles.

This thesis focuses on the proposal of hybrid concept of cuckoo search with firefly algorithm (CS-FAPP) for the optimal path planning. These nature inspired meta-heuristic algorithms are considered due to the adaptability of these concepts according to problem definition. Meta-heuristic algorithms are population based global optimization concepts that work iteratively on the experience sharing manner to improve the existing solutions of NP hard problems. Further, the selection of firefly algorithm and cuckoo search over other meta-heuristic algorithms are the advantages of these algorithms over other concepts. Firefly algorithm is global optimization based algorithm inspired from flashing behavior of fireflies that can provide optimal solution in various constrained and unconstrained optimization problems. The superiority of the firefly algorithm in different fields is observed by the work presented by Yang (2009) [109], Gandomi et al. (2011) [110] and Fister et al. (2013) [111].

Although, there is higher esteem of firefly algorithm but its usability in path planning field is rarely observed. The optimum solution and performance results of firefly algorithm in existing work [109-111] motivate us to consider it in path planning.

As firefly algorithm is population based meta-heuristic algorithm so it is necessary to maintain the proper stability among the exploitation and exploration of concept to solve any problem. Exploitation can be defined as the process to find solution space nearby to discovered regions in exploration. Exploration is the process to discover new regions in the defined workspace. The improper balance among two can trap firefly agents in the local optimum. In firefly algorithm, this balance can be maintained with the parameters of light absorption coefficient and randomization parameter. But there may be a chance that individual firefly algorithm can be trapped in local optimum and fails to find the overall global solution for optimal path planning problem. To overcome this drawback, cuckoo search is used along with the firefly algorithm as cuckoo search works as an individual agent with brood parasitic behaviour and levy flight property. The motive to consider the concept of cuckoo search for the application of path planning is existing applicability of cuckoo search for path planning [112]. These factors of cuckoo search and firefly algorithm motivate us for hybridization of concepts for better path planning results.

In this hybrid approach of CS-FAPP, the levy flight property of cuckoo search has been used by the firefly algorithm in which multi-agents work on the problem domain to find the optimized path. Moreover, the obstacles of the pathway are handled by considering the obstacles as the worst nest for the egg. After handling the obstacles, firefly agents move randomly with levy flight property and reach the destination with best possible path from defined initial point to end point.

1.7. RESEARCH OBJECTIVES

The research topic of this thesis is "**Analysis of Nature Inspired Intelligence in the Domain of Path Planning and Searching with Consideration of Various Parameters**". To achieve this goal, following research objectives has been structured:

1. To find the optimized path in cross country using proposed hybrid algorithm of Firefly and Cuckoo search algorithm.
2. To evaluate the performance of proposed hybrid algorithms by altering their parameters like distance, communication, behaviour and weather conditions in them.
3. To compare the results with the existing hybrid approach of BAT and Cuckoo Search in finding optimized path using standard benchmark functions.

1.8. RESEARCH CONTRIBUTION

This thesis work presents a hybrid approach of CS-FAPP algorithm for the identification of optimized path in a static environment. The proposed algorithm is the amalgamation of the SI based concepts: firefly algorithm and cuckoo search. This proposed hybrid approach handles the obstacles in the path way and identifies the optimized path from initial defined source point (S). The experimentation is performed different terrain features with major focus on vegetation land cover of the different Indian regions. The proposed algorithm combines the properties of both firefly and cuckoo search algorithms. The multi agent firefly algorithm employs the property of CS algorithm for the imitation of the host egg's pattern which is used to handle the obstacle in the path and levy flight property to find the solutions after each iteration. The key contributions of the present work are:

- A hybrid approach of CS-FAPP algorithm for the identification of optimized from considered initial start point to final end point of workspace area by handling the obstacles in the pathway.
- The consideration of benchmark functions for the performance testing of proposed hybrid CS-FAPP algorithm.

- The implementation of the proposed approach on Google based satellite images of different Indian regions for its applicability in real world scenarios.
- Evaluation and comparison of the proposed approach with the hybrid CS with BA (CS-BAPP) and individual BA, FA, and CS on the basis of evaluation parameters of number of iterations, error rate, success rate, and simulation time.

1.9. Thesis Organization

The thesis entitled is organized in the form of five chapters. The work is initialized with the current introduction chapter (**Chapter 1)** which presents the introduction of thesis by presenting the overview of research work, basic concepts related to Nature inspired computing, its different types, swarm intelligence techniques, path planning, applications of path planning, types of different environment, path planning techniques, research objectives, problem statement and motivation behind the consideration of present work. Also, the basic fundamentals of CS, FA, and BA algorithm are explained in this chapter. Moreover, the procedure of path planning using individual concepts of CS, FA, and BA are also discussed in this chapter.

Further, **Chapter 2** brings the existing work related to optimal path planning. This chapter also presents the tabular representation of existing path planning techniques with tabular entries of obstacle type, considered environment type, and system type.

Chapter 3 elucidates the proposed Research Methodology of CS-FAPP used for optimal path planning. As there is the need to compare the results of proposed hybrid algorithm (CS-FAPP) with hybrid CS and BA (CS-BAPP), so modified algorithm of hybrid CS and BA (CS-BAPP) is also presented.

Chapter 4 presents the validation results of proposed hybrid algorithm CS-FAPP by evaluating the benchmark functions for the CS-FAPP. The benchmark functions of CS-BAPP are also evaluated and compared with CS-FAPP. Comparison of these hybrid algorithms based on benchmark functions is also presented with individual results of CS, FA, and BA. It also discusses the experimental setup, information related to MATLAB simulation software, and used database.

This chapter also presents the detailed description of the procedure followed to perform the path planning using CS-FAPP by considering the working example with defined obstacles. The results of the proposed CS-FAPP are assesses based on the evaluation parameters of success rate, simulation time, error rate, and number of iterations. The chapter is ended presenting the comparison of used research methodology with CS-BAPP and individual CS, BA, and FA.

Chapter 5 concludes the thesis work based on the evaluated results and discussion along with some future directions.

CHAPTER 2
RELATED WORK

This chapter covers the existing techniques and methods used by researchers for optimal path planning. Different authors have used different method, obstacle types, and environment for the path planning. The literature review of the existing techniques is divided into three sections. First section presents the work related to optimal path planning using swarm intelligence concepts. Second section presents the existing work on path planning using popular concepts other than swarm intelligence techniques. Third section presents the work related to combination of both the swarm intelligence based concepts and other concepts for path planning. The work of three sections is discussed as follows:

2.1. PATH PLANNING USING SWARM INTELLIGENCE

In this section, the work related to swarm intelligence based path planning concepts are investigated. The popular algorithms investigated are Particle Swarm Optimization (PSO), Bacterial Forging Optimization (BFO), Bat Algorithm (BA), Firefly Algorithm (FA), Ant Colony Optimization (ACO), and Cuckoo Search (CS) etc. Table 2.1 illustrates the work of several authors using swarm based path planning techniques. The considered research work in year-wise pattern is discusses here.

Qin et al. (2004) [113] implemented the concept of PSO for the mobile robot path planning. The working of algorithm was divided into two phases. In the first phase, Dijkstra's shortest path algorithm is applied to find the shortest path from the specified source to the goal point. In the second phase, PSO algorithm is implemented with the motive to optimize the results obtained during the first phase. The shortest path obtained from Dijkstra approach is optimized to determine the optimized route with the help of PSO. An attempt to improve PSO using mutation operator is also made to remove the local minima problem. The system has shown efficient performance results.

Wang et al. (2006) [114] also developed a mathematical model to handle the obstacles in dynamic environments. The author implemented a soccer robot system based on PSO algorithm by examining the shape of the soccer robot. A PSO based coding mechanism is proposed and the parameters including search space, length of the binary string are considered to evaluate the computation time. The obtained simulation results for total 50 times implementations of algorithm, 48 times optimal results are achieved and 2 times algorithm suffer from local minima.

Kang et al. (2008) [115] proposed the mathematical model for the implementation of Dijkstra algorithm and PSO. The author initially developed an improved Dijkstra algorithm to uncover the shortest path from specified initial point to termination point. A graph is formed using MAKLINK and an attempt is made to lessen the length of the path using the property of triangular inequality. Further, the shortest path identified by improved Dijkstra algorithm is optimized by PSO. In the experimentation, the path obtained from three algorithms: Dijkstra, improved Dijkstra and improved Dijkstra with PSO, indicate that the length of path obtained from improved Dijkstra with PSO is minimum.

Zhao et al. (2009) [116] further attempt to modify PSO to avoid obstacles in a real-time dynamic environment. The traditional PSO is improved by considering the polar coordinates and knowledge information about the domain of path planning. The proposed polar coordination PSO (PPSO) algorithm works in two steps of local and global path planning. Initially, the information of static obstacles is acquired to obtain the global optimal path. Then, the mobile robot predicts the local optimal path by estimating the moving position of dynamic obstacles along with the global optimal path. The experimental results obtained from PPSO are compared to traditional PSO and Genetic algorithm clearly indicates that PPSO outperforms and is applicable in real-time environment. Effective navigation in rough terrain is one of the real-time path planning applications.

ZakeriNejad et al. (2010) [117] implemented PSO to determine the route for vehicle in the rough terrain. The authors developed a control mechanism to guide and supervise the vehicle along the path generated by the PSO. The experimentation is conducted in two different environments to make sure the obstacle handling and real-time applicability of the algorithm. In the first scenario, the motion of the vehicle begins from a hole in the middle of ground. This situation tests the applicability of PSO algorithm in dynamic and real time situations. The vehicle has to move either back and forth or has to swirl in order to acquire momentum. In the second scenario, number of walls with side ramps is considered to ensure the collision avoidance property of PSO algorithm. The results evaluated in both experiments favored PSO for path planning in rough terrain. Real-world scenarios consist of several situations like the generation of fire in some rescue mission or landmines etc. Robot must evade these situations during the progression of determine the optimized route.

Kundra and Sood (2010) [118] implemented the combination of two swarm based BBO and PSO algorithms for cross country path planning of autonomous outdoor vehicles. The author implemented PSO to identify the path from specified source to goal position by detecting the obstacles in the path. The information retrieved from PSO is used by BBO algorithm to avoid the detected obstacles and continue the movement towards the goal point. The simulation results evince the efficiency of proposed algorithm.

Zhu et al. (2011) [119] presented an improved ant colony based algorithm to uncover a collision free optimal path with static/dynamic obstacles within unknown environmental conditions. The algorithm is named as Robot Navigation Ant algorithm (RNA) which is emulated from the food searching procedure of scout ants. The initially marked target node is the node closest to the outer boundary of the robot visual domain. The robot is provided with sensors to gather the information about the obstacles and to construct a mapped model in its range. Then, the Multi Scout Ant Corporation (MASC) method is implemented to uncover the static path in the present local map i.e. within the range of robot visual domain. Here, only static obstacles are considered. After planning the local path, the robot looks for the moving obstacles and identifies the changes of collision of obstacles in the identified static path. In case, an unavoidable collision is detected, then moving obstacle is assumed as static obstacle and again MASC is implemented to identify new local static path. The entire process is repeated until the robot identifies the actual destination point in its visual domain. The simulation results indicate the viability of algorithm even in complex environmen-t with moving obstacles.

Englot and Hover (2011) [120] have proposed the extension of ant colony framework for the application of TSP problem. Authors have specifically mentioned the combination of point to point sampling approach with ACO algorithm and applied for the determination of optimal multi-goal path planning. Authors have employed the proposed approach for the application of underwater vehicle based goal detection. In this approach, there is the availability of different obstacles information and goals information. Overall, results are determined with metric total mission time which is the function of planned mission duration and computational cost. The modification of ACO algorithm is to detect the next goal is considered as the priority task instead of physical path movement by ants itself.

Liu et al. (2012) [121] designed an adaptive firefly algorithm for path planning in a known environment. The author attempted to ameliorate the performance of standard firefly algorithm by adjusting its parametric values with number of iterations. Initially, the environment information along with the beginning and ending points is attained. The firefly population is generated and the brightness value of each firefly is obtained. The fireflies are compared in terms of brightness with the neighboring fireflies and the values of random & absorption parameters are updated. The brightest firefly indicates the best path belongs to the brightest firefly. The simulation results are obtained in a workspace region of 450*350 with 9 static obstacles.

Sood and Kaplesh (2012) [122] proposed another cross country path planning approach by using combination of two swarm intelligence techniques. The authors modified the previously proposed algorithm by implementing bee colony optimization (BCO) technique for obstacle avoidance instead of BBO. Natural paths from specified source to destination points are identified by PSO and then BCO is implemented to avoid detected obstacles and to optimize the identified path. The successful implementation of the proposed algorithm motivates the author for further experimentation.

Sood and Kaur (2012) [123] implemented BBO and BCO for the cross country path planning. BBO is implemented to identify the paths and obstacles from specified source to destination. Whereas, the detected obstacles are avoided and further the identified path is optimized by using BCO.

Wang et al. (2012) [124] attempts to implement cuckoo search optimization algorithm in the domain of uninhabited combat air vehicle (UCAV). The authors have used the concepts of cuckoo search with differential evolution to determine an optimized route for the air vehicles. The cuckoo search algorithm is serving the purpose to identify the optimal path and differential evolution further improves the cuckoo search algorithm by optimizing the selection criteria of cuckoos when a nest is updated. For the random selection of the next mutation and crossover is considered instead of the levy flight property for the selection of the next nest. In levy flight, the nest is selected randomly but the selection criterion is optimized by mutation and crossover property of differential evolution. The simulation results proved the feasibility of proposed algorithm for UCAV path planning.

Wang et al. (2012) [125] modified the original bat algorithm with mutation and presented the concept of Bat Algorithm with Mutation (BAM) for the application of UCAV problem. In this approach, the proposed BAM algorithm determine the efficient path by the selecting the appropriate nodes with minimum cost function and handling the treat fields. Authors have evaluated and compared the result outcomes with other possible optimization approaches like SGA, PSO, ES, GA, BBBO, ACO etc. Authors noticed the superiority of results in comparison with other mentioned concepts. Authors also mentioned the future issues related to path planning concepts such as collaborative route determination for UCAVs and self adaptive route determination by single UCAV.

Wang et al. (2012) [126] further presented the modified firefly algorithm (MFA) for the application of UCAV path planning. Authors compared the results with other concepts of original bat algorithm and other optimization concepts such as SGA, PSO, ES, GA, BBBO, ACO etc. Overall results self elaborates the effectiveness and efficacy of MFA. The MFA algorithm also compared with existing methods on the basis of different dimensions and maximum generation. Most of the cases present the dominance of proposed concepts. Future issues of collaborative route determination for UCAVs and self adaptive route determination by single UCAV are also highlighted in the paper.

Chen et al. (2013) [127] developed a two stage ACO algorithm for robotic path planning. The author split the process into two stages: pre-processing and path planning stage. The algorithm is based on the standards of scent pervasion. Initially, the scent is broadcasted in the environment map and the ants search for the broadcasted scent according to the "1-minus" search strategy. The ants are declared as "dead ants" after reaching the destination point and their corresponding path length is calculated. Once all the ants are reached at the destination point, then their corresponding path length is observed and best solution with shortest path length is selected. The comparison of the two stage ACO algorithm with other algorithms showed the efficiency of the approach in different scenarios.

Zhang et al. (2013) [128] developed a multi-objective PSO based path planning algorithm for robot in the presence of inadequate environmental conditions. Initially, a global path for the robot is described by considering the fuzzy membership function with risk factor and path distance as parameters. The risk degree is not fixed rather it defines a range of values for the moving obstacles. The problem is named as a bi-objective optimization problem which is resolved by implementing multi-objective PSO algorithm. In multi-objective PSO the feasibility paths are improved by random sampling of particles and uniform mutation. The real-time applicability of mentioned algorithm in uncertain environmental scenarios is evaluated by considering four different test case scenarios. The results obtained the viability of multi-objective PSO in uncertain environmental scenarios.

Mohanty and Parhi (2013) [129] have proposed a new variant of cuckoo search algorithm to solve the path planning problem for mobile robots. Authors have considered the environment with variant obstacle types and unknown environmental conditions. The major benefit of CS algorithm is the requirement of lesser parameters for simulation in comparison with PSO and GA. In this application, the route planning problem is converted into minimization problem and objective function is selected as per the target position. The results of proposed algorithm indicate to achieve the target position with path length in comparison with PSO and GA.

Mohanty and Parhi (2014) [130] have again used the CS algorithm for the path planning of mobile robots with the workspace of static obstacles and partial known or unknown environmental conditions. The obstacles of the pathway are determined by nest position functionality of host nest. The task of the experiment is to determine the shortest path from defined initial point to end point (destination). The results of the proposed algorithm are evaluated in terms of path length. The comparison results of proposed algorithm in comparison with existing GA and PSO approach indicates the superiority of the CS algorithm.

Montiel et al. (2015) [131] implemented the bacterial potential field (BPF) optimization algorithm for the planning of optimal path for mobile robots. The proposed algorithm aims to identify safe and optimal path with static and dynamic obstacles. The algorithm is integrated with an artificial potential field method to remove its shortcomings and to acquire an optimized path. Before the navigation of mobile robot, BPF is called to identify the initial path. Mobile robot follows the point to point process and moves further. BPF is called whenever the environment experiences a change i.e. when obstacles are added or removed. Thus, the algorithm works offline as well as in online mode. The computer simulation environment developed for performance evaluation of proposed algorithm evinces its feasibility and effectiveness.

Das et al. (2016) [132] presented a combinational approach of IGSO (Improved Gravitational Search Optimization) and IPSO (Improved Particle Swarm Optimization) for the planning of optimal path from defined initial source point to end destination point. The key motive of the proposed hybridized algorithm is to obtain the optimized results with minimum energy consumption, minimize path deviation and path length. In this experimentation, static obstacle types and environmental conditions are considered with dynamic robots. The results of hybrid IPSO-IGSO algorithm are compared with hybrid HMBO-Tabu list, hybrid GSO-PSO, individual

IGSO, and individual IPSO algorithm through Khepera-II and simulation. Overall efficient results are achieved with proposed hybrid IPSO-IGSO algorithm. The achieved results indicate the dominance of proposed hybrid IPSO-IGSO algorithm.

Zeng et al. (2016) [133] further have used the improved concept of PSO with differential evolution and non-homogeneous Markov chain. The considered concept of switching local evolutionary PSO (SLEPSO) is experimented for intelligent robot path planning in grid based system. Two environmental conditions are considered for the testing of concept. In the first grid, there are lesser obstacles and another one contains more number of obstacles. Overall, improved results using SLEPSO are reported in comparison with other PSO based considered concepts.

Liu et al. (2016) [134] have used the PSO algorithm for planning of optimal path with unmanned aerial vehicle (UAV). In this case, obstacles are static and environmental conditions are known. Overall efficient results are reported for path planning in comparison with standard particle swarm optimization, firefly algorithm, and genetic algorithm.

Ayari and Bouamama (2017) [135] have presented the improvement on PSO algorithm by considering it with dynamic distributed condition and named as dynamic distributed PSO (D^2PSO). The proposed algorithm is multiple robots based algorithm in which particle of PSO acts as robots and each robot try to obtain the optimal path from defined source point to destination point in known environmental conditions and obstacle types. Here, two optimal detectors are LOD_{gbest} and LOD_{pbest} are considered where, LOD_{gbest} is considered for global best and LOD_{pbest} is considered for particle best. If any robot unable to achieve any of the global best and local best results, then the robotic particle are replaced with re-structured particles following some repetitive iterations. The proposed algorithm achieved the optimal path length with minimum possible conditions.

Bibiks et al. (2018) [136] have improved the cuckoo search algorithm to apply on the application of resource constrained project scheduling algorithm. This optimization of scheduling can be applied to the work of planning of optimal path. The authors have improved the concept of cuckoo search to apply on the discrete scheduling problem by changing the key elements like solution improvement, representation scheme, and Levy flight. Overall authors have improved the results in comparison to state of art considered concepts.

Patle et al. (2018) [137] have used the firefly algorithm for the planning of optimal path for mobile robot navigation in uncertain environmental conditions. The experimentation is performed on unknown environment type with changing obstacles from static to dynamic and environmental conditions. The experimentation is performed and observed with multiple and single robots of different 10 and 20 rounds. Authors also observed the simulation & experimental results and noticed the lesser deviation of around 5.7% with same path length and simulation time.

Goel et al. (2018) [138] have worked for the route planning of unmanned aerial vehicles using the SI based Glow-Worm Swarm Optimization (GWSO) concept. Authors have conducted three experiments with three types of environments of static obstacles with static gorals, random static obstacles with static goals and dynamic obstacles with moving goals. The increasing complexity of environment leads to increase in simulation time and cost of the project. Overall, authors suggested considering the work to solve the real life problems such as flood affected areas and other disaster times.

Table 2.1: Path Planning using Swarm Intelligence Techniques

Author and Year	**Method**	**Obstacle Type**	**Environment Information**	**System Type**
Qin et al., 2004 [113]	PSO	Static	Known	Mobile robot system
Wang et al., 2006 [114]	PSO	Dynamic	Known	Soccer robot system
Kang et al., 2008 [115]	PSO	N/A	Known	Mobile robot system

Author and Year	**Method**	**Obstacle Type**	**Environment Information**	**System Type**
Zhao et al., 2009 [116]	PPSO	Static and dynamic	Known	Mobile robot system
ZakeriNejad et al., 2010 [117]	PSO	Static	Known	Vehicular system
Kundra and Sood, 2010 [118]	Hybrid PSO and BBO	Static	Unknown	Autonomous outdoor vehicle
Zhu et al., 2011 [119]	RNA and MASC	Static and dynamic	Unknown	Mobile robot system
Englot and Hover, 2011 [120]	ACO	Static	Known	Hovering Autonomous Underwater Vehicle
Liu et al., 2012 [121]	FA	Static	Known	Mobile robot system
Sood and Kaplesh, 2012 [122]	Hybrid PSO and BCO	Static	Unknown	Autonomous outdoor vehicle
Sood and Kaur, 2012 [123]	Hybrid BBO and BCO	Static	Unknown	Autonomous outdoor vehicle
Wang et al., 2012a [124]	CS	N/A	Known	Uninhabited combat air vehicle
Wang et al., 2012b [125]	BA	N/A	Known	Uninhabited combat air vehicle
Wang et al., 2012c [126]	FA	N/A	Known	Uninhabited combat air vehicle
Chen et al., 2013 [127]	ACO	N/A	Known	Mobile robot system

Author and Year	Method	Obstacle Type	Environment Information	System Type
Zhang et al., 2013 [128]	PSO	Dynamic	Unknown	Robot system
Mohanty and Parhi, 2013b [129]	CS	Static	Unknown	Mobile robot system
Mohanty and Parhi, 2014 [130]	CS	Static	Unknown	Mobile robot system
Montiel et al., 2015 [131]	BPF	Static and dynamic	Known	Mobile robot system
Das et al., 2016a [132]	Hybrid PSO and GSO	Static	Known	Multi-robot system
Zeng et al., 2016 [133]	SLEPSO	Static	Known	Intelligent Robot system
Liu et al., 2016 [134]	PSO	Static	Known	Unmanned Aerial Vehicular System
Ayari and Bouamama, 2017 [135]	D^2PSO	Static	Known	Multi-Robot system
Bibiks et al., 2018 [136]	Improved Discreate CS	Static	Known	Resource Constraint Project Scheduling
Patle et al., 2018 [137]	FA	Static and Dynamic	Unknown	Mobile Robot Navigation System
Goel et al., 2018 [138]	GWSO	Dynamic	Known	Unmanned Aerial Vehicles

2.2. PATH PLANNING USING CONCEPTS OTHER THAN SWARM INTELLIGENCE

In this section, the work related to popular concepts other than swarm intelligence algorithms used by different authors for path planning concepts are discussed. The popular algorithms considered are Tabu Search (TS), Fuzzy Logic (FL), Simulated Annealing (SA), Genetic Algorithm (GA), and Neural Network (NN) etc. Table 2.2 illustrates the work of several authors based on concepts other than swarm intelligence concepts for path planning techniques. The considered research work in year-wise pattern is discusses here.

Althoefer et al. (2001) [139] developed the approach for real-time robot navigation. The approach is developed by implementing fuzzy rules with the combination of reinforcement learning neural network. This navigation system is entirely based on local information and the distance between obstacles & target configuration. The distance and angular velocity between the nearest obstacle and current link are evaluated to find the one local optimal link path. The experimental result shows a powerful system for robot navigation.

Sauter et al. (2002) [140] considered a pheromone field to guide the robotic vehicles in the battle space including target point and threats. The robotic vehicle is guided to find the safe entry and exit path avoiding different threats in the field. The experimentation was conducted by considering six different test cases implemented on evolution strategies and genetic algorithm. The basic mechanism includes five different components of the pheromone field: place, surrogate (red), walker (blue), ghost and flavor. The physical field space is represented in hexagonal shape by place agents with neighbors. The red surrogate agents represent the threats and enemy targets. The blue walker agent is our friendly unmanned robotic vehicle. Now, the ghost agents are created by the walker agents to inspect the battle field and to develop a route from the walker to the specific mark. The surrogate, walker and ghost agents deposit some amount of pheromone named as flavors. The agents are distinguished based on their flavor of pheromone deposit. Ghost agents wander the entire battle field in search of route from the walker to target position. In case of success, the ghost returns to the walker and deposits its pheromone which is further used to determine an ultimate shortest route from the walker to target. The experimental results witnessed the success of genetic algorithm in all six test cases and also conclude that genetic algorithm constructs a stronger path when compared with evolution strategy.

Park and Lee (2002) [141] further integrated simulated annealing with artificial potential field (APF) approach for the path planning of mobile robots. Path planning with APF approach usually results into local minima. So, the author integrated artificial potential field with SA approach to run off from the local minima. The efficacy of the system is analyzed by comparing the results of integrated approach with the APF only. The APF algorithm was trapped in local minima whereas the integrated approach escaped from it efficiently.

Hamdan and Hawary (2003) [142] implemented the genetic algorithm as an application of multicast routing. The networking space was considered as an undirected weighted graph with edges & nodes. The string of genes is evaluated indicating the number of paths from the specified beginning point node to final point node. The cost of each path is evaluated and total cost of all paths defines the delay. The simulation results are evaluated on network space with different set of nodes and the obtained outcomes work in the favor of proposed method.

Lee and Wu (2003) [143] implemented fuzzy logic to navigate the mobile robot with obstacles in an unknown environment. Initially, the mobile robot determines its orientation and position along with the environment information. The heading direction with highest priority is determined from the proposed fuzzy algorithm and based on the information obtained from sensors. As a result, the robot reached to the intermediate location and the same process is iterated at each intermediate location until robot reaches to destination.

Erickson (2003) [144] implemented an ANN for robot navigation in a dynamic environmental conditions. Neural network models the relationships between input and output layers for path planning task. No learning procedure is required by the network and the neurons are connected to each other according to the geometry of the search space. The target neurons and obstacles are represented as set of neural activities.

Neural activities include attraction and repulsion where target neurons are globally attracted and obstacle neurons are locally repulsive. The network model has shown reliability against the change in conditions without giving any additional knowledge.

Janglová (2004) [145] developed the mobile robot path planning, a two neural network model. The network model plans the motion of the mobile robot in a static workspace. The first neural network is learned to determine the vacant segment. The find-space problem is taken under consideration to deliver safe output path. The first network model is developed by considering principle component analysis (PCA) network, combining the supervised and unsupervised learning procedure in the same network topology. The second NN is a multilayer perceptron network model, developed to conclude the route for mobile robot. The generation of collision free paths in the simulation results demonstrates the efficiency of network model.

Hu and Yang (2004) [146] attempted to resolve the problem of robotic path planning using problem-specific GA (knowledge based) rather than standard genetic algorithm. The proposed algorithm is designed with the potential to perform small scale local search and also exhibit domain specific knowledge which makes it suitable for both static and dynamic environmental conditions. The computational time computed from the simulation results indicates the practical usage of knowledge based genetic algorithm.

Loo et al. (2004) [147] enhanced GA approach for path planning by traversability vector approach. The problem is formulated by considering the polygon shaped obstacles in the search space. In the next step traversability vector approach is implemented for collision detection. The t-vectors are constructed with three possibilities for the specific path segment: path segment with no obstruction, possible obstruction and definite obstruction. Then, the genetic operator is applied and fitness function of every path segment is identified. The feasible and infeasible paths are inspected and the corresponding optimal path is identified by analyzing the fitness function values.

Wu et al. (2004) [148] have developed a vector based fuzzy logic for underwater robotics vehicle motion planning. A simulated ocean scenario is considered for the robotic vehicle where robotic vehicle efficiently utilized the orientation and direction parameters for its motion planning. The methods of fuzzification, reasoning and defuzzification are constructed for the fuzzy path planner by defining a set of motion planning rules. The evaluated results evince the proficiency of the proposed fuzzy logic based algorithm. The task of path planning is challenging, but in case of already known environmental information, the task to acquire obstacle information becomes comparatively easy. In contrast, the lack of environmental information makes path planning more challenging.

Wang and Liu (2005) [149] attempted to implement fuzzy logic in an unknown environment. The authors called it "blind goal-oriented navigation" as an autonomous mobile robot reaches to goal point without any prior information about the search space. A minimum risk approach is introduced to plan path locally and to handle the problem of local minima. The process includes path searching, obstacle avoidance, and goal seeking. The fuzzy logic efficiently handles the coordination and the path searching mechanism.

Masehian and Amin-Naseri (2006) [150] developed an online motion planner to control and supervise the movements of the mobile robot system. The robot system is instructed with the target position and each iteration involves set of actions (tabu rules) based on the current robot location. The whole process is similar to visibility graph method as mobile robot is attracted towards the vertices representing the obstacles. The cost function and sensor information are determined to find the hurdles in the path and to construct optimized route from the source to target. The experiment results have shown the feasibility of tabu search for robotic path planning problems.

Zhu and Yang (2006) [151] developed a neural network for task assignment to a multi-robotic system in a dynamic workspace. The network is developed based on self organising map learning process to plan the motion of the robot integrated with task assignment. Since the system is multi-robot system, numbers of robots dynamically adjust their motion according to the assigned tasks. The proposed network model adjusts automatically with the change in environmental conditions even in case of uncertainties. The simulation results evince the efficient working of the network model.

Hui et al. (2006) [152] have also considered a neuro-fuzzy approach for a car type mobile robot system. A fuzzy logic controller (FLC) is proposed to manage the car mobile robot. The author has implemented three different neuro-fuzzy techniques for the improvement of the fuzzy logic controller. First Approach follows the default rule to move with zero deviation and maximum acceleration. The second approach is fuzzy logic controller based manually constructed approach with a five layer feed forward neural network. Unlike rule based neuro-fuzzy system in second approach, third consists of a neuro-fuzzy system with back propagation learning. The three approaches are compared with each other in several computer simulations and neuro-fuzzy approach has shown significant results.

Araujo (2006) [153] introduced another neuro-fuzzy system for robotic navigation in a dynamic environment. The author aims to develop mobile robot based navigation system to construct their own maps in dynamic and unknown environment. A system named PAFARTNA which stands for "Prune-Able fuzzy ART Neural Architecture" is introduced. The proposed architecture is integrated with a navigation mechanism enabling the system to build navigation maps even in dynamic environment scenarios.

Lilly (2007) [154] introduced the obstacle avoidance in the process of path planning. He elaborated a fuzzy based obstacle avoidance mechanism for the autonomous vehicles. The learning of the obstacle controller is provided by some of negative regulations. Initially, the positive regulations are acquired which are based on the expert knowledge that is already known. The fuzzy logic is learnt with the negative rules in collaboration with the positive defined rules. The simulations show effective obstacle avoidance results in different test case scenarios.

AL-Taharwa et al. (2008) [155] presented an idea to implement genetic algorithm in a static and controlled environment. The search space is initially converted into grid shaped graph with nodes. The obstacles at the nodes of the grid are already defined. The fitness function between the two nodes is computed and these values of fitness function helps to determine the shortest path which is considered as the optimal path from defined initial-point to end-point.

Bi et al. (2008) [156] presented another mobile robot navigation system for unknown and dynamic environment. The authors implemented a two layer hierarchical approach of genetic algorithm and fuzzy logic. Initially, a global path is identified in the first layer by implementing genetic algorithm and then in the second layer fuzzy is implemented to find the local and optimized path for a mobile robot. The input of fuzzy system is portioned into four sets namely: goal seeking, right edge following, left edge following, and obstacle avoidance. The behavior of the sets is learned by the system and the final output results into a local optimized path. The evaluated simulation results indicate the favorable results to proposed algorithm.

Engedy and Horváth (2009) [157] developed an artificial neural network (ANN) to plan a path for the robotic car in the workspace in which there is the known information of moving and static obstacles. Obstacle handling in path planning problem is a crucial and primary step. The robot is installed with a camera to observe the workspace area. The images taken from the camera are processed by image processing algorithm to track the information of the robot position obstacles and the destination position. The back propagation learning based neural network is trained with the controller data for the identification of the path for robot. Also, the motion of the robot is controlled by gateway hardware with basic commands of change in speed or stopping the vehicle. The author concluded that in case of moving obstacles with low speed, it is possible to train model online to promote its applicability in real-time scenarios.

Parhi and Singh (2009) [158] presented navigation control of mobile robot in real-time scenarios. A four layer back propagation learning network model procedure is developed to identify the safe path by avoiding moving and static obstacles. The mobile robot motion is initially analyzed and the next step includes the analysis of neural network. The neural network input is target angle along with the information of left, right and front distance with obstacle. The mobile robot detects the presence of obstacle based on sensor information. The absentia of obstacle in the acquired sensor information leads the mobile robot to seek target information. The comparison of results is made with the fuzzy controller and both models have shown a good agreement.

Chao et al. (2009) [159] introduced fuzzy logic based obstacle handling and target spot tracking system. The mobile robot works in three primary modules. In the first module, visual information is gathered. The mobile robot is configured with two CCTV cameras to gather the target and obstacle information. The second module is a strategy decision module. This module results in an executing strategy along with fuzzy controller demands. The third module i.e. motor driver module receives the commands as input and then decodes the commands to reach to the target destination. The evaluated results manifest the viability of the used approach.

Cheng et al. (2010) [160] presented another combination of the characteristic properties of exact algorithms with meta-heuristic algorithm. A dynamic programming and genetic algorithm based path planner is implemented for autonomous under water vehicles. The problem is formulated by considering two different seabed landscape models with different underwater scenarios. Initially, *n* number of crossover points will be generated and then located randomly in the first generation. A B-spline path segment is constructed by randomly selecting any two crossover points. Then deterministic crossover operator transforms the random crossover operator to a classic genetic algorithm crossover operator. The fitness value of all the crossover operators will be stored in a fitness matrix which will be sorted to find out the best path. The result outcomes of proposed algorithm and genetic algorithm indicate the proficiency of proposed algorithm.

Alajlan et al. (2013) [161] recently, investigated the path planning capability of genetic algorithm in large scale search space. The algorithm is initializing with a random population and every single element represents a feasible path. Then, greedy approach is considered for the identification of initial route from start position to the destination point. A set of intermediate cells are randomly selected to build a new path from beginning point to end point. The value of fitness function is evaluated at each step until the population size reaches its maximum limit. The obtained values are ordered for the identification of optimal shortest path. The computed results are compared with the A* algorithm shows the equality to both algorithms in almost all cases.

Duan and Huang (2014) [162] proposed a hybrid method based on Imperialist Competitive Algorithm (ICA) and Artificial Neural Network for path planning in Unmanned Combat Aerial Vehicle (UCAV). The ICA based ANN optimization of the path starts with the network initialization parameters and membership functions. The network based root mean square error is evaluated to select the imperialists and then imperialists are moved to their respective colonies based on fitness function values. The weakest colony with the lowest fitness value is eliminated and the remaining value is updated. The output is the optimal path from the specified source to destination. The results obtained from the proposed optimization model are compared with ABC algorithm results which evince that the considered hybrid approach is efficient than ABC for 2-D path planning.

Chaari et al. (2014) [163] investigated path planning based on tabu search algorithm in grid environment. The mobile robot navigates in a grid partitioned into cells with same size. The position of the obstacles is already known and obstacles are static in nature. Tabu search iteratively search for the optimal feasible path till the completion of termination criterion. The algorithm is assumed with three basic moves: insert, remove and exchange. The applicability of move either improves the current solution or will deteriorate the path solution and path at each iteration is checked for its feasibility. After a number of consecutive steps, the search process stagnates. Then diversification is considered to move the process to a novel search space. The proposed algorithm evinced promising results in comparison with the A* algorithm.

Behnck et al. (2015) [164] modified a simulated annealing algorithm for the small unmanned aerial vehicles to plan their path. The author aims to propose a calculation algorithm for calculating flight paths in already mentioned situations. In this, each UAV is transitioning on given region with several points of interest (POI) and each POI must be visited by an UAV. The problem is alike the TSP where each city must be visited once. Now, simulated annealing algorithm solves the problem by either moving the POI of one path to the other path or by swapping the POIs of the same path. The algorithm orders the path and modifies the larger paths as compared to the smaller ones. The simulation results were favorable to proposed algorithm.

Lu et al. (2016) [165] have used the deep convolutional neural network for the path planning for NAO bio-mimetic robot in dynamic and static environments. The obtained results in terms of various evaluation parameters indicates the dominance of deep convolutional neural network in comparison with particle swarm optimization, support vector machine, and back propagation neural network.

Lee and Kim (2016) [166] focused on the population initialization method in a conventional GA and proposed a modified-GA with effective initialization method. In the enhanced initialization method, the search space is divided into grid of directed acyclic graph with *n* nodes and edges. The input to the graph is starting and destination nodes and the output is the set of *n* number of paths. After the initialization, the graph is created with new feasible edges. The new feasible edges are identified by implementing the Bresenham's line algorithm. The graph thus created identifies the best path from specified starting and destination point. The identification of high quality obstacle free path favours the success of proposed algorithm over genetic algorithm.

Ni et al. (2017) [167] considered the bio-inspired neural network (BINN) for the path planning with underwater mobile robot system. The proposed system needs not any learning for the system and environment. Overall efficient solution of path planning has been obtained even in the existence of floating obstacles in underwater environment but the unknown information of environment lower down the efficiency of system.

Lin et al. (2017) [168] in the recent years have improved the GA based compliant robot path planning (GACRPP) by improvement in the population initialization using improved bidirectional rapidly-exploring random tree (Bi-RRT) approach. In this approach, initially the connections between clusters are established using improved Bi-RRT, then population is initialized with back tracking and final smoothen path is obtained using improved genetic algorithm. Authors reported the improved performance of path planning using improved genetic algorithm approach.

Chen et al. (2017) [169] have proposed the novel concept of fuzzy support vector machine (FSVM) based on general regression neural network (GRNN) for the improvement of anti-jamming ability for autonomous vehicle. In this concept, initially A* algorithm is used to determine the negative and positive samples. Then, training process is performed using FSVM approach in which membership functions is determined based on GRNN. Finally, obtained path is smoothened using Bezier interpolation algorithm. Authors concluded the proposed GRNN-FSVM approach is efficient to determine the collision free path by handling the static obstacles in outdoor environment.

Bayat et al. (2018) [170] have used the charged particles based electrostatic potential field (EPF) approach for the mobile robots based path planning. The entire concept is based on the EPF approach with the implementation of work in discrete environment type.

Table 2.2: Path Planning using Concepts Other than Swarm Intelligence Techniques

Author and Year	**Method**	**Obstacle Type**	**Environment Information**	**System Type**
Althoefer et al., 2001 [139]	Combination of Neural Network and Fuzzy Logic	Static and dynmaic	Known	Multi-robot system
Sauter et al., 2002 [140]	Genetic Algorithm	Dynamic	Known	Unmanned robotic vehicle
Park and Lee, 2002 [141]	Simulated Annealing	Static	Known	Mobile robot system

Author and Year	Method	Obstacle Type	Environment Information	System Type
Hamdan and Hawary, 2003 [142]	Genetic Algorithm	N/A	Known	Network system
Lee and Wu, 2003 [143]	Fuzzy logic	Static and Dynamic	Known	Mobile robot system
Erickson, 2003 [144]	Artificial Neural Network	Dynamic	Known	Planner robot
Janglova, 2004 [145]	Neural Network	Static	Known	Mobile robot system
Hu and Yang, 2004 [146]	Genetic Algorithm	Static and dynamic	Known	Mobile robot system
Loo et al., 2004 [147]	Genetic Algorithm	Static	Known	Mobile robot system
Wu et al., 2004 [148]	Fuzzy logic	Dynamic	Known	Underwater robotic vehicle
Wang and Liu, 2005 [149]	Fuzzy logic	Static and dynamic	Unknown	Mobile robot system
Masehian and Amin-Naseri, 2006 [150]	Tabu Search	Static	Known	Mobile robot system
Zhu and Yang, 2006 [151]	Neural Network	N/A	Known	Multi-robot system
Hui et al., 2006 [152]	Neuro-fuzzy	Dynamic	Unknown	Car type Mobile robot system

Author and Year	Method	Obstacle Type	Environment Information	System Type
Araujo, 2006 [153]	Neuro-fuzzy	Dynamic	Unknown	Mobile robot system
Lilly, 2007 [154]	Fuzzy logic	Static and dynamic	Unknown	Autonomous vehicle
AL-Taharwa et al., 2008 [155]	Genetic Algorithm	Static	Known	Mobile robot system
Bi et al., 2008 [156]	Genetic Algorithm and Fuzzy Logic	Dynamic	Unknown	Mobile robot system
Engedy and Horváth, 2009 [157]	Neural Network	Static and Dynamic	Known	Robotic car
Parhi and Singh, 2009 [158]	Neural Network	Static and Dynamic	Known	Mobile robot system
Chao et al., 2009 [159]	Fuzzy logic	Static	Known	Mobile robot system
Cheng et al., 2010 [160]	Genetic Algorithm	N/A	Known	Autonomous under water vehicles
Alajlan et al., 2013 [161]	Genetic Algorithm	N/A	Known	Mobile robot system
Duan and Huang, 2014 [162]	Neural Network	N/A	Known	Unmanned combat aerial vehicle

Author and Year	Method	Obstacle Type	Environment Information	System Type
Chaari et al., 2014 [163]	Tabu Search	Static	Known	Mobile robot system
Behnc et al., 2015 [164]	Simulated Annealing	N/A	Known	Small unmanned aerial vehicle
Lu et al., 2016 [165]	Neural Network	Static and Dynamic	Known	NAO bio-mimetic robot
Lee and Kim, 2016 [166]	Genetic Algorithm	Static	Known	Mobile robot system
Ni et al., 2017 [167]	Neural Network	Static and Dynamic	Unknown	Underwater robotic vehicle
Lin et al., 2017 [168]	Genetic Algorithm	Static and Dynamic	Known	Compliant Robot System
Chen et al., 2017 [169]	Combinational approach of Fuzzy logic and Neural network	Static	Unknown	Autonomous off-road vehicle
Bayat et al., 2018 [170]	EPF Approach	Static	Known	Mobile robot system

2.3. PATH PLANNING USING COMBINATION OF SWARM INTELLIGENCE AND OTHER CONCEPTS

This section discusses the existing concepts related to combination of popular concepts of swarm intelligence and other than swarm intelligence algorithms. Table 2.3 illustrates the work of several authors based on the combinational concepts. The considered research work in year-wise pattern is discusses here.

Min et al. (2005) [171] have performed system simulation of a mobile robot based on PSO approach to determine a global optimal route in a complex and dynamic environment. Initially, a second order mathematical model is proposed for the obstacle avoidance. The parameters including the velocity of mobile robot, direction of the present obstacle and the destination point are observed to propose the obstacle avoidance mathematical model. Then, the PSO is implemented to ensure the optimal obstacle free path. To obtain the precise results, the authors modified the PSO using the crossover and mutation concept of Genetic algorithm. The author considered different environmental scenarios to compare the modified PSO algorithm with APF algorithm and modified APF using GA (APF-GA). The environmental scenarios contain single static obstacles, multiple static obstacles and multiple dynamic & static obstacles. The comparative simulation result outcomes clearly indicate the outperformed performance of improved PSO in all environmental scenarios.

Mohamad et al. (2005) [172] implemented another combination of meta-heuristic technique with classical techniques. The authors amalgamate the concept of ACO with Probabilistic Road Map (PRM) method to identify path for a mobile robot. The multi-goal problem is handled by constructing a road map with edges and nodes in a C-shape workspace. The authors have implemented Lazy algorithm for the identification of the route from specified source point to end point and multi-goals are handled by considering the same approach as it is used in traveling salesman problem. A global table is maintained to identify the global optimal path. Each ant communicates with each other and also works independently to identify the route from the nest to source of food. Every time a particular ant drops a trail and the values at global table were modified. The experimental results are evaluated by considering four different case scenarios. The calculated results are in the favor of the considered concept.

Mohamad et al. (2006) [173] proposed another algorithm inspired from ACO and PRM in a 3-dimensional robot path planning. In the initial stage a road map is structured from the specified beginning to target point. Then the nodes and edges in the road map are replaced by the foraging ants. The next stage introduced trail-ants and then finally the multi-goal problem is handled by applying ACO to the traveling salesman problem approach. A global table is maintained throughout the entire process and whenever an ant drops a trail, its value is updated. The evaluated results with parameters of number of intermediate points and time taken for the planning, work favorably for the proposed algorithm.

Mei et al. (2006) [174] hybridized the concept of ACO with classical method APF approach to identify the optimal route from defined initial point to end point without any collision in a dynamic environment. The algorithm works in two phases: global planner and local planner. Initially, a global path is identified by ACO algorithm in static environment. Then, the information acquired by global planner is used to execute local planner phase. In the local planner phase, the real-time dynamic obstacles are avoided and local optimal path is identified by implementing the artificial potential field algorithm. The simulation results obtained after comparing ACO with genetic algorithm clearly indicates the superiority of ACO.

García et al. (2007) [175] presented a simple ACO based meta-heuristic algorithm (SACO-MH) for robot navigation and path planning in a dynamic environment. For the optimal path planning, the navigation search space is considered as a graph matrix of 50*50 nodes. The selection of the next node depends on the Euclidean distance of the current node from the node in a 3*3 window and it depends on the evaluated fuzzy cost function. The fuzzy inference system based output gives the total weight of each ant. The algorithm is capable of handling dynamic obstacle. In case of obstacle in the path, the mobile robot identifies the new path from the previous position. The proposed algorithm has shown promising results.

García et al. (2009) [176] further continued their work and proposed another ACO based algorithm named SACOdm proposed. In this algorithm, the decision making ability of the mobile robot depends on the distance among the initial node and target node. Hence, the name SACOdm suggests simple ant colony optimization where *m* for memory and *d* stands for decision. The memory feature is introduced in order to keep track of the previously visited nodes. The optimal path selection depends on the cost function evaluated in fuzzy inference system by implementing simple tuning algorithm. The results in the form of global optimized path evaluated during the experimentation are in the favor of proposed concept.

Huang and Tsai (2011) [177] developed another hybrid mobile robot system of meta-heuristic algorithms. The author combined the advantages of PSO and GA for collision free global path planning. In the algorithm, PSO operates the crossover and mutation properties of genetic algorithm to identify the local best solution and the global optimal solution from the set of local best solutions. Further, B-Spline smoother is implemented to remove any type of discontinuities. As a result, a collision free smooth global path from specified source to destination is obtained. Robotic path planning using hybrid GA and PSO algorithms in grid environment usually consider fixed number of turning points with individual population values.

Ju et al. (2014) [178] proposed to eliminate the issue in PSO and GA with hybridization based approach. A tree structure is considered for the path planning where initially all the paths based on fitness value and their respective binary trees are created. Then, PSO is applied to enhance the paths and crossover operator of GA is considered to create new paths. The new paths created are modified using mutation operator and the whole process continues until an optimal path is retrieved. In comparison to the grid based methods, fewer numbers of turning points are taken by the proposed approach.

Contreras-Cruz et al. (2015) [179] recently introduced an integrated technique of evolutionary programming (EP) with ABC algorithm for mobile robot path planning. The algorithm works in two main steps. In the initial step, an initial path from specified beginning to end spot is identified by ABC algorithm. The local path identified is then optimized globally by evolutionary programming. The efficiency of algorithm is investigated against probabilistic road map technique which proves the efficiency of meta-heuristic technique over classical technique.

Panda et al. (2016) [180] proposed a multi-robot path planning system. The author implemented the hybrid tabu search and PSO for the path planning. Initially, PSO is implemented to search for the paths locally and then the particles are operated using tabu search to acquire the optimal path. The neighbourhood solutions and their respective cost functions are calculated to acquire one best feasible solution. The experimental results work in the favor of proposed technique.

Das et al. (2016) [181] have proposed the hybridization of differential perturbed velocity (DV) and Improved PSO (IPSO) algorithm for the planning of optimal path for multi-robot system. In this hybrid approach, DV algorithm is used to manage the velocities of robots. However the considered obstacle types and environmental conditions are static in nature, the considered robots are dynamic in nature. The results of the proposed hybrid algorithm are compared with individual differential evolution algorithm, individual DV algorithm, and individual IPSO algorithm through Khepera-II and simulation. The overall evaluated results indicate the superiority of proposed hybrid IPSO-DV algorithm.

Zhang et al. (2018) [182] have modified the bare bone PSO algorithm with the integration of modified differential evolution (DE) algorithm. There was the consideration of three main indices of path safety degree, smoothness degree, and path length for the concept of path planning. Authors noticed the outperformed results on the basis of mentioned metrics.

Table 2.3: Path Planning using Combination of Swarm Intelligence Techniques and Other Techniques

Author and Year	Method	Obstacle Type	Environment Information	System Type
Min et al., 2005 [171]	Hybrid PSO and GA	Static and dynamic	Unknown	Mobile robot system
Mohamad et al., 2005 [172]	Hybrid ACO and PRM	Static	Known	Mobile robot system
Mohamad et al., 2006 [173]	Hybrid ACO and PRM	Static	Known	Mobile robot system

Author and Year	**Method**	**Obstacle Type**	**Environment Information**	**System Type**
Mei et al., 2006 [174]	Hybrid ACO and APF	Dynamic	Known	Mobile robot system
García et al., 2007 [175]	ACO and Fuzzy logic	Dynamic	Known	Mobile robot system
García et al., 2009 [176]	ACO and Fuzzy logic	Static and dynamic	Known	Mobile robot system
Huang and Tsai, 2011 [177]	PSO and GA	Static	Known	Mobile robot system
Ju et al., 2014 [178]	PSO and GA	Static	Known	Mobile robot system
Contreras-Cruz et al., 2015 [179]	ABC and EP	Static	Known	Mobile robot system
Panda et al., 2016 [180]	PSO and tabu search	Static	Known	Multi-robot system
Das et al., 2016b [181]	Hybrid IPSO and DV	Static	Known	Multi-Robot system
Zhang et al., 2018 [182]	Hybrid PSO and DE	Static	Known	Mobile Robot system

2.4. ANALYSIS AND DISCUSSION

The overall analysis is presented based on the three categories cateroies of considered concepts of path planning using swarm intelligence, other than swarm intelligence concepts and combinational approaches. The analysis has been presented on the basis of publication distribution, Overall distribution of major categories used for path planning, distribution of obstacle types, and distribution of environmental types. This analysis is made based on the considered research publication of path planning.

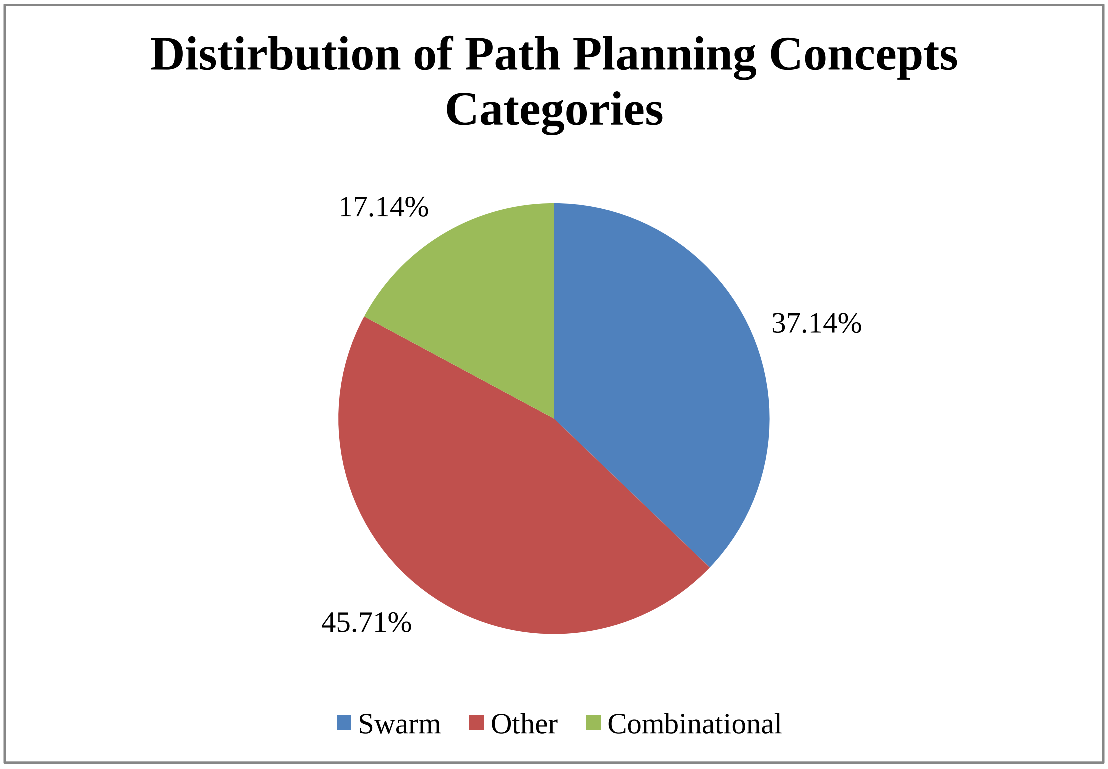


Figure 2.1: Distribution based on Path Planning Concepts Categories

Figure 2.1 presents the distribution of publications based on major categories of considered concepts for path planning. The selected quality contributions of researchers are discussed by considering techniques into three categories of swarm intelligence based concepts and other than swarm intelligence based concepts and combinational approaches. In figure 2.1, the word ‘swarm’ indicates the publication of path planning based on swarm intelligence concepts, word ‘other’ indicates the publication of path planning based on other than swarm intelligence concepts, and the word ‘combinational’ indicates the publication of path planning based on combination of swarm intelligence and other concepts.

The figure 2.1 indicates the 37.14% research contributions are based on individually swarm intelligence techniques, 45.71% research contributions are all the other categories of path planning techniques, and 17.14% research contributions considers combinational techniques based on both the swarm intelligence and other computational intelligence concepts. Although the other category have attained the value 45.71% which is greater than swarm intelligence concepts, but the other concepts category includes the machine learning concept, soft computing approach (other than swarm intelligence), and other traditional algorithms. It indicates the more proportion of swarm intelligence concepts as compared to any other category.

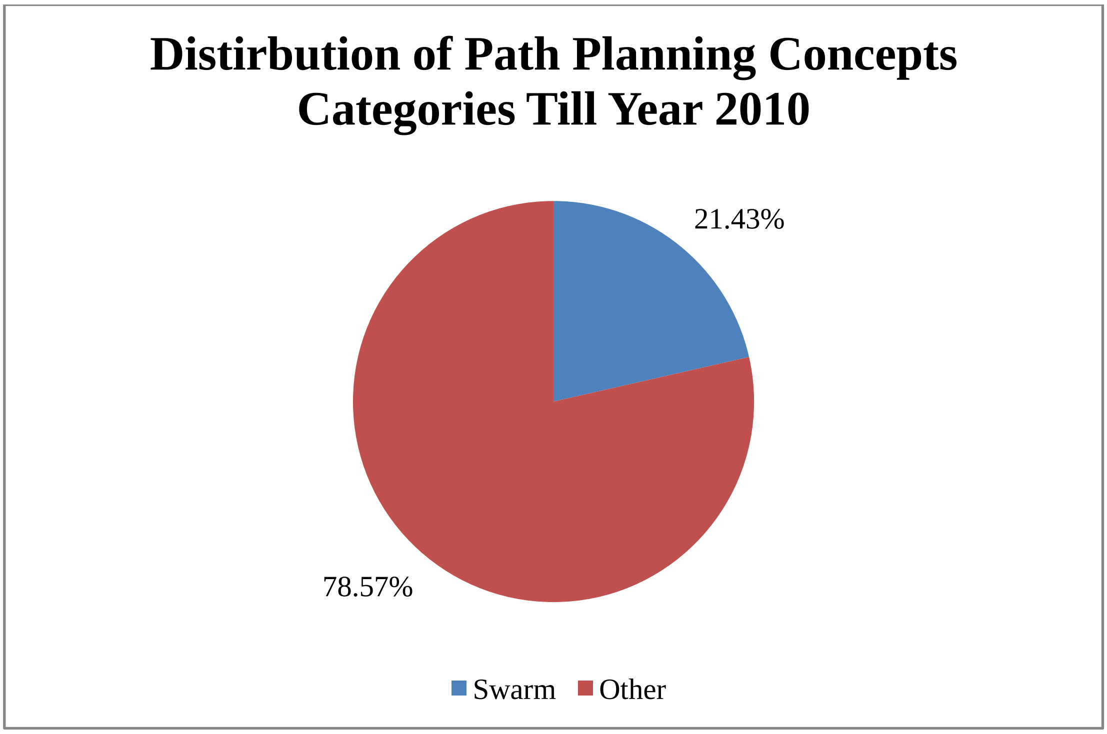


Figure 2.2: Distribution of Path Planning Concepts Categories till Year 2010

Further research publication distribution is presented based on the publications till the year 2010 and after the year 2010. For this distribution, the combinational concepts category is not considered as it contains both types based on swarm intelligence and other concepts. This research distribution till year 2010 is presented in figure 2.2 and research distributions after year 2010 are presented in figure 2.3.

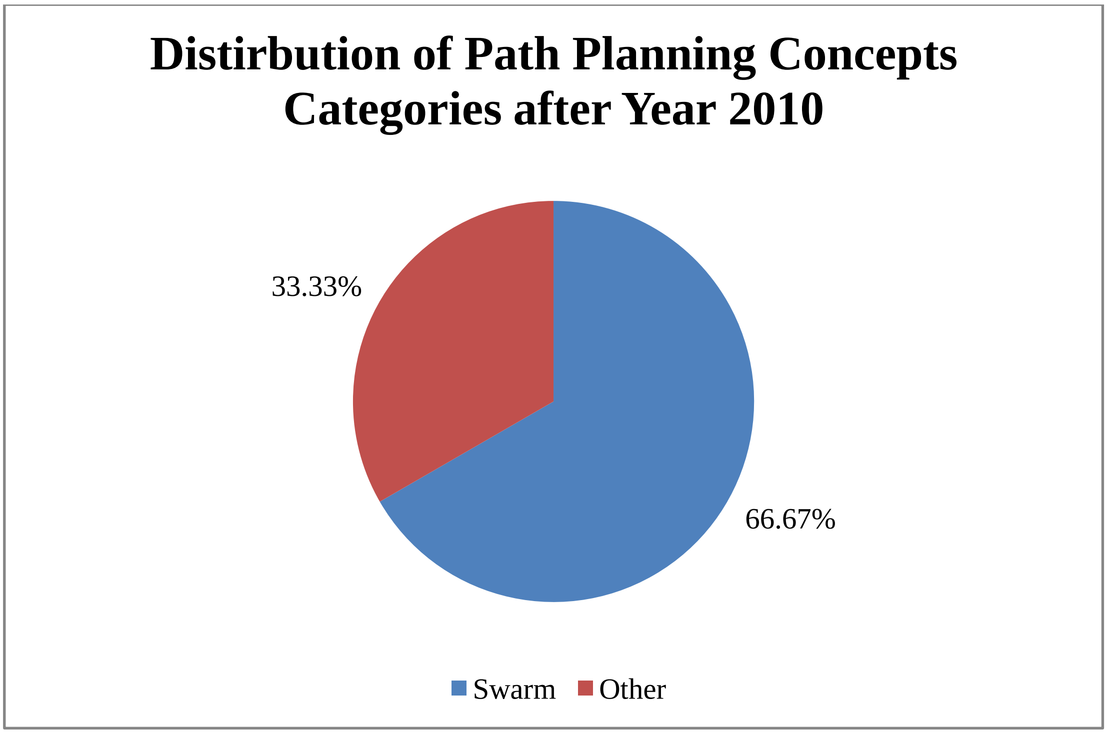


Figure 2.3: Distribution of Path Planning Concepts Categories after Year 2010

The figure 2.2 indicates only 21.43% research contribution of swarm intelligence concepts till year 2010 and 78.57% research contributions of all the other concepts till year 2010. This indicates the greater use and importance of concepts other than swarm intelligence techniques till year 2010. After the year 2010, the usability and adaptability of swarm intelligence concepts changed as presented in figure 2.3. A great rise in the usage of swarm intelligence concepts for path planning is observed with 66.67%. There are only 33.33% techniques in which other concepts are used for path planning. It indicates the growing adaptability of swarm intelligence techniques for path planning.

Among swarm intelligence techniques, major work on PSO, BA, FA, CS, ACO, and BPF algorithms is discussed. In this research work, our major focus is on the FA, CS, and BA. Although, there are lesser research quality publications on FA, BA, and CS, we tried to consider the efficient and related publications. All the three algorithms are efficient to work for path planning applications but the individual concepts face the problem of trapping in local optima, and need more simulation time. This obligates us to hybridize the considered concepts.

On the other hand, techniques other than swarm intelligence techniques studied in this work consists of genetic algorithm, fuzzy logic, simulated annealing, tabu search, and neural network. There is a lot of applcability of neural network in path planning due to its powerful parallel processing, learning ability, and non-linear mapping characteristics. But the technique is very requires a lot of training data for its effective working which makes it time consuming process. The processing of hidden layers in neural network is a black-box processing, so it is often very inconvenient to determine number of layers and their respective number of neurons. Fuzzy logic is a powerful technique with an ability to stimulate human mind and represent it with the help of linguistic variables. It is a knowledge based technique with IF-THEN rules, but the membership function and selection of rules are very hard to determine. Genetic algorithm is the popular meta-heuristic techniques and is successfully considered in path planning problem. Genetic algorithm is easy to implement but one of the primary downside in genetic algorithm is its applicability in grid map and no such mechanism to control population diversity which makes the algorithm unsuitable for dynamic environments. The algorithm suffers from multiple local optima and does not support multi-objective problems. Further, both tabu search and simulated annealing are powerful meta-heuristic techniques with a strong ability to escape from local minima. However, the parameter selection is a difficult task as there is no problem specific parameter selection procedure.

The combination of swarm intelligence based techniques and other techniques have also attained efficient results in different environments and different concepts are presented by different authors. Although combinational approaches are comparable to swarm intelligence concepts but the hybridization of two swarm based algorithms provides better results.

Further the distribution of contributions based on types of obstacles handled during path planning is presented. This distribution of work is presented in figure 2.4. For this, we have considered three categories: Static, Dynamic, and Both. During this distribution, we have not considered the research contributions whose obstacle handling information is not available.

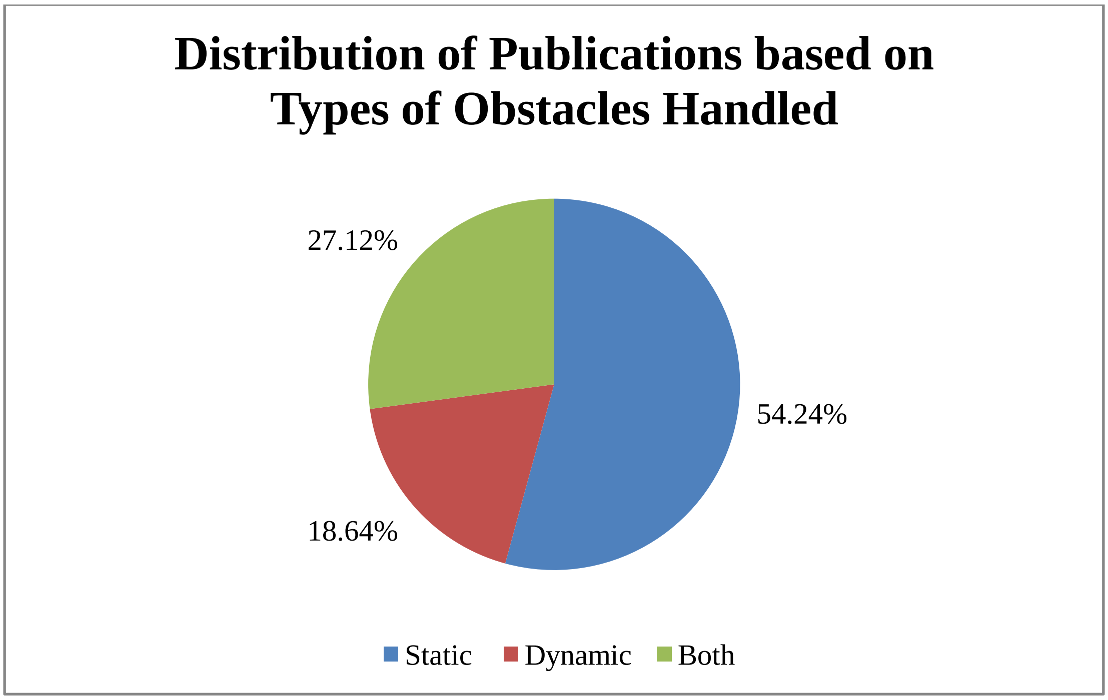


Figure 2.4: Distribution of Publications based on Types of Obstacles Handled

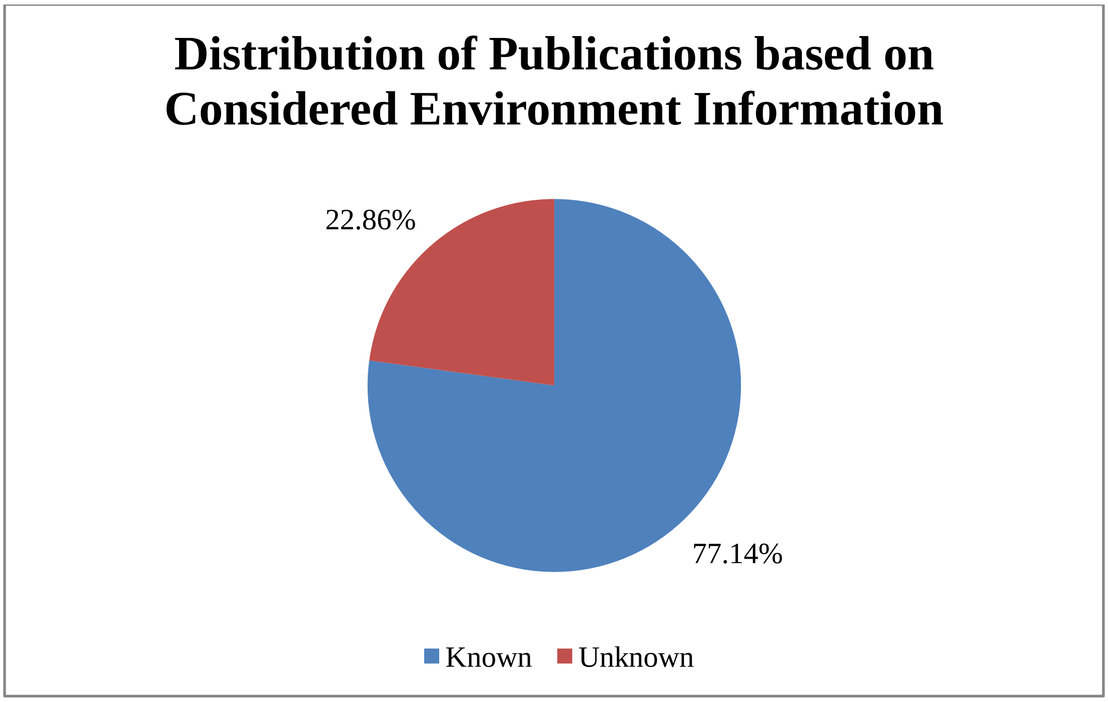


Figure 2.5: Distribution of Publications based on Environment Information

On the basis of research contributions in which obstacle information is available, it can be noted from the figure 2.4 that 54.24% authors have worked only on the static obstacle types, 18.64% authors have worked on only the dynamic obstacle types, and 27.12% authors have handled both the static and dynamic obstacles for the path planning. From figure 2.4, it can be declared that there are more research contributions based on static obstacles handling.

Further the distribution of research contributions is presented based on types of environment considered by authors to do the path planning. This distribution of work is presented in figure 2.5. There are two types of environment information either known or unknown. During this distribution, we have again not considered the research contributions whose environment information is not available.

The figure 2.5 indicates the 77.14% authors have worked on the path planning concepts in which environment information was known to them. But there are also the authors who have worked on the path planning concept by considering unknown environment information (22.86%). In the proposed research work, we have considered the unknown environment type with static obstacles in the path of workspace database.

2.5. SUMMARY

This Chapter provides a comprehensive knowledge of path planning with existing work from 2001 to 2018. The center stone of this chapter is the analysis of swarm intelligence and other popular techniques for path planning. The considered concepts are classified into three categories which are: swarm intelligence based techniques, other than swarm intelligence techniques, and combination of swarm intelligence and other techniques. Moreover, the work on basic fundamentals of popular swarm intelligence concepts is also discussed.

Next chapter discusses the proposed Research Methodology of CS-FAPP used for optimal path planning. The comparable concept of hybrid CS and BA (CS-BAPP) is also presented in next chapter.

CHAPTER 3
RESEARCH METHODOLOGY

This chapter focuses on the proposed hybrid concept of CS-FAPP which is hybrid approach of Cuckoo Search and Firefly Algorithm. Moreover, the hybrid concept of CS and BA which is termed as CS-BAPP is also discussed in this chapter as the algorithm of proposed CS-FAPP is compared with the CS-BAPP algorithm. This chapter is presented in two sections of Hybrid CS-FAPP and Hybrid CS-BAPP.

3.1. HYBRID CS-FAPP

It is very restrictive to solve optimization problem within decent time interval using single swarm intelligence algorithm. The process of hybridizing/combining one swarm intelligence concept with other swarm intelligence algorithm results into a time efficient algorithm with high flexibility. In this research work, the two meta-heuristic algorithms i.e. cuckoo search and firefly algorithms are hybridized to determine the optimum and collision free route from specified source and destination in a static and unknown environment. This algorithm is especially modified to solve the path planning problems. The proposed algorithm combines the properties of the two algorithms to detect the obstacles present and to find the best path from source point to destination point by handling the obstacles in the pathway. In this research work, obstacles are the terrain surfaces such as rocky areas, urban livings houses, vegetation areas especially trees, water surface (as it can't be crossed without swimming features) and other manmade non-living objects. By handing these mentioned obstacles, the proposed CS-FAPP approach can route the path on the plain surface, lower grass vegetation, and barren surfaces. The major focus of this work is to plan the path in the vegetation terrain regions by handing the obstacles the mentioned obstacles.

The multi-agent firefly algorithm performs the local and global search to find the best path. The fireflies use the pattern mimicking property of cuckoo bird to find the obstacle present and the concept of levy flight is used to randomly move the fireflies. To further elaborate this statement, initially all the fireflies are considered to find the optimal solution. These fireflies moves are based on the property of levy flight.

Further, during the travel of fireflies from sources to destination, if any obstacle found, then these obstacles are handled with cuckoo search based property with an assumption of considering obstacle as worst nest for cuckoo egg. In this way, fireflies use the property of pattern mimicking to handle the obstacles. After, handling the obstacle, fireflies again proceed towards the destination by calculating the Euclidean distance from the neighbouring pixels to destination point.

In the proposed algorithm, the workspace is considered as collection of binary pixels values: 0 and 1 where value 1 indicates obstacle free white pixel and value 0 is black pixel with obstacle. Initially, it is assumed that *n* number of fireflies contain *n* number of candidate solution and one amongst them is the global optimal solution. The firefly population is initialized starting from source point and one firefly which is the brightest is placed at destination point in order to attract other fireflies. The fireflies use the pattern mimicking property of cuckoo bird to detect and handle obstacles by neglecting the obstacle point as worst nest for cuckoo egg. When the obstacle is detected then the Euclidian distance of the neighbour pixels from destination point is calculated to find the next point for the movement of firefly agents. If p_i and dp_j are pixel point and destination point respectively then the Euclidean distance between them can be evaluated by using Equation (3.1).

$$d_{ij} = \|p_i - dp_j\| = \sqrt{\sum_{k=1}^{n} (p_{ik} - dp_{jk})^2}$$

… Equation (3.1)

The neighbour point with minimum distance is selected for further processing. Due to the attractiveness property of fireflies, fireflies moves from the source to destination as brightest firefly is assumed to be placed at destination. The attractiveness β of a firefly is proportional to its brightness and it decreases with increase in distance d. Thus, attractiveness of a firefly is proportional to light intensity $L.I.$ and it can be calculated with the help of Equation (3.2):

$$\beta = \beta_0 \, e^{-\gamma d^2}$$

… Equation (3.2)

Where β_0 is the attractiveness at $d = 0$ and γ is the light absorption coefficient.

The steps of the proposed hybrid algorithm are listed here:

1. Initialize source point sp and destination point dp.
2. Initialize light absorption coefficient γ and initial attractiveness β_0 at 0 value.
3. Initialize the workspace as set of pixel values i.e. $p(i) = (p_1, p_2, p_3 \ldots, p_n)$.
4. Calculate Objective function $f(x)$ $where\ x = (x_1, x_2, x_3, \ldots \ldots x_n)^t$ for each firefly.
5. Initialize the population p of fireflies and each firefly possesses a candidate solution.
6. Search for the path and the obstacles present in the path using the pseudo code mentioned below:

While $(t < maxgeneration)$ or $(stop\ criteria)$ **do**
for $(i = 1{:}n)$ do
// check for presence of obstacle using cuckoo's pattern mimicking property
if$(p_i == 1)$ then
for $(j = 1{:}n)$ do
If $(L.I_j > L.I_i)$
Move firefly i towards j via levy flights
endif
Calculate attractiveness according to equation (3.2)
Update light intensity value and evaluate new solution
end for j
else
Obstacle is detected
Calculate Euclidean distance of neighbour pixels using equation (3.1)
Proceed with nearest pixel point
Again check for the presence of obstacle
end for i
order fireflies to find current best solution
end while

7. The path identified is collision free optimum path from specified source to destination.

As proposed CS-FAPP algorithm is designed for the path planning from source point to destination point in a static and unknown environment. In this research work, Google based satellite images of different regions of India are considered (Discussed the database in chapter 4) which consists of different terrain features for experimentation. The work flow of path planning using proposed hybrid CS-FAPP algorithm is presented in figure 3.1. The detailed stepwise explanation of algorithm for path planning with more description is discussed below:

Step 1: Consider the input satellite image and define the source s_p and destination d_p for the path planning.

Step 2: Apply Morphological operation to reduce the blocked paths and unnecessary area gaps between the source to destination.

Step 3: Initialize the parameters of firefly algorithm for n-fireflies possessing random position x_i in d-dimensional search space with initial threshold brightness value. The other parameters used are intensity value of fireflies I, initial attractiveness β_0, at initial position, and light absorption coefficient γ.

Step 4: Firefly works on the fundament to attract other firefly. Attractiveness of firefly is always determined in terms of their Brightness (Intensity) value which is related to the objective function. Brightness of firefly obeys the inverse square law where brightness decreases as the distance between two firefly's increases. This can be evaluated as shown in Equation (3.3).

$$I \propto 1/r^2$$

…Equation (3.3)

Where, I is the intensity value and r is the distance.

Step 5: To evaluate the distance between the two firefly i and j, apply the Cartesian distance formula as shown in equation (3.4).

$$r_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$$

…Equation (3.4)

Where, r_{ij} is the distance between i and j firefly, and x, y are the coordinates.

Step 6: Calculate the attractiveness value for each pixel (firefly) of the image using the objective function below as shown in equation (3.5).

$$\beta = \frac{\beta_0}{1 + \gamma r^2}$$

…Equation (3.5)

Where, β is the attractiveness, β_0 is the attractiveness at distance r = 0 and γ is the light absorption coefficient.

Step 7: Fireflies move randomly with the function of attraction towards the brighter one. Initially, each firefly search for random solution. Then, obtain the optimized global solution by sharing their information with other fireflies.

Step 8: The obstacles are handled with cuckoo search based property with an assumption of considering obstacle as worst nest for cuckoo egg. In this way, fireflies use the property of pattern mimicking to handle the obstacles.

Step 9: When the obstacle is detected, then the Euclidian distance of the neighbor pixels from destination point is calculated to find the next point for the movement of firefly agents. If p_i and dp_j are pixel point and destination point respectively, then the Euclidean distance between them can be evaluated by using equation (3.6).

$$d_{ij} = \|p_i - dp_j\| = \sqrt{\sum_{k=1}^{n} (p_{ik} - dp_{jk})^2}$$

… Equation (3.6)

After, handling the obstacle, fireflies again proceed towards the destination by calculating the Euclidean distance from the neighboring pixels to destination point.

Step 10: Evaluate the firefly algorithm based global optimum solution for all the pixel points in the way from source to destination using step 4 to step 9.

Step 11: Obtain the final shortest optimum path from source to destination.

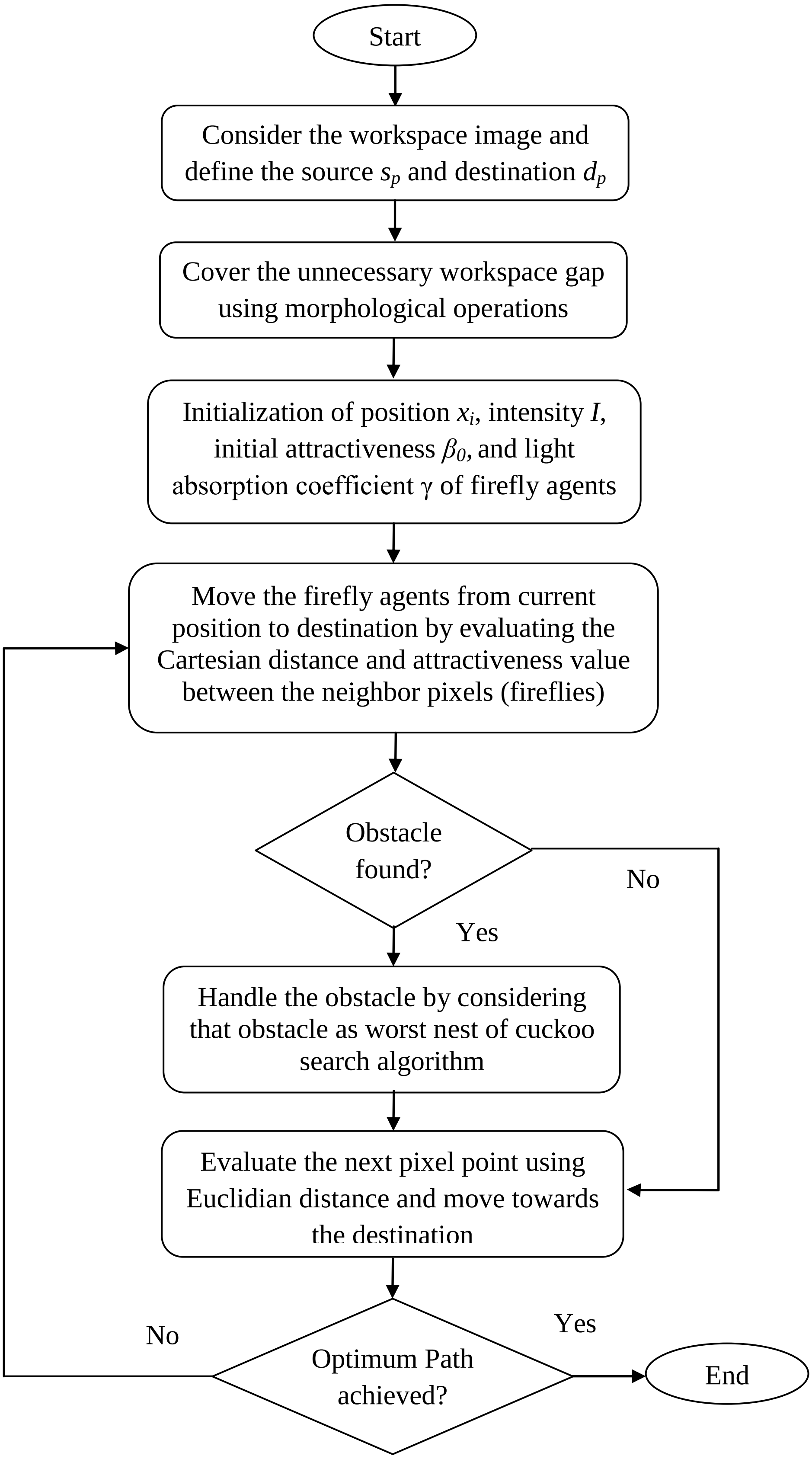


Figure 3.1: Workflow of path planning using proposed hybrid CS-FAPP

The above mentioned steps are used to find the optimal path from defined source to destination by handing the obstacles in the way. The proposed hybrid CS-FAPP algorithm is evaluated by making the comparison with another hybrid algorithm of CS-BAPP. The modified hybrid algorithm of CS-BAPP is discussed below for the path planning.

3.2. HYBRID CS-BAPP

This sub-section presents the proposed hybrid concept of CS-BAPP to find the optimal path from source to destination. As discussed above, the proposed hybrid CS-BAPP algorithm combine the properties of the two swarm intelligence based algorithms to handle the present obstacles and to find the optimal path from source point to destination point in a static and unknown environment. This algorithm is also designed for the path planning application with experimentation on the Google based satellite images of different regions of India which consists of various terrain features. The considered workspace region is considered as collection of binary pixels values: 0 and 1 where value 1 indicates obstacle free white pixel and value 0 is black pixel with obstacle. In this hybridized approach, BA uses the pattern mimicking property of cuckoo bird to handle the obstacle present and the concept of levy flight is used to randomly move the bats. The work flow of path planning using proposed hybrid CS-BAPP algorithm is presented in figure 3.2. The step by step algorithm is discussed here:

Step 1: Consider the input satellite image and define the source s_p and destination d_p for the path planning.

Step 2: Apply Morphological operation to reduce the blocked paths and unnecessary area gaps between the source to destination.

Step 3: Initialize the parameters of bat algorithm for n-bats possessing random position x_i in d-dimensional search space. The other parameters used to move the bats are velocity v_i, minimum frequency f_{min}, maximum frequency f_{max}, constant pulse rate r, and loudness A.

Step 4: Evaluate the size g_{best_i} and position x_{best_i} for the minimum fitness function value. The fitness function value can be evaluated using Equation (3.7).

$$f(x_i) = \sum_{k=1}^{D+1} P_{k-1}P_k$$

…Equation (3.7)

Where, $i=1,2,\ldots n$, $P_{k-1}P_k$ stands for distance between two adjacent nodes P_{k-1} and P_k, and n is the number of bats. The value of P_k can be evaluated using equation (3.8).

$$P_i = P_i^{(0)} + \left(P_i^{(1)}P_i^{(0)}\right)h_i$$

…Equation (3.8)

Where, h_i is the scale parameter that belongs to [0, 1] for $i = 1, 2, \ldots, D$. Here, D stands for number of nodes, $P_i^{(0)}$ and $P_i^{(1)}$ are the endpoint of nodes.

Step 5: Further, evaluate the frequency f_i and update the velocity v_i^t and position x_i^t using the equation (3.9) to equation (3.11).

$$f_i = f_{min} + \beta(f_{max} - f_{min})$$

…Equation (3.9)

$$v_i^t = v_i^{t-1} + (x_i^t - x^*)f_i$$

…Equation (3.10)

$$x_i^t = x_i^{t-1} + v_i^t$$

…Equation (3.11)

Where, β is uniform and random distribution factor lie between [0, 1], x^* is the globally best solution among the entire population. In this research work, f_{max} is considered as 100 and f_{min} is considered as 0.

Step 6: If the value of random number is larger than the constant pulse rate r, then bats will follow the random walk process as evaluated with equation (3.12).

$$x_{new} = x_{old} + \in A^t$$

… Equation (3.12)

Where, A^t is the average loudness value of all the bats at any time t and the value of random number $\in$ lie between [-1, 1].

Step 7: The obstacles are handled with cuckoo search based property with an assumption of considering obstacle as worst nest for cuckoo egg. In this way, bats use the property of pattern mimicking to handle the obstacles.

Step 8: When the obstacle is detected then the Euclidian distance of the neighbour pixels from destination point is calculated to find the next point for the movement of bat agents. If p_i and dp_j are pixel point and destination point respectively then the Euclidean distance between them can be evaluated by using equation (3.13).

$$d_{ij} = \|p_i - dp_j\| = \sqrt{\sum_{k=1}^{n} (p_{ik} - dp_{jk})^2}$$

… Equation (3.13)

After, handling the obstacle, bats again proceed towards the destination by calculating the Euclidean distance from the neighboring pixels to destination point.

Step 9: Evaluate the bat algorithm based global optimum solution for all the pixel points in the way from source to destination using step 4 to step 8.

Step 10: Obtain the final shortest optimum path from source to destination.

3.3. SUMMARY

In this chapter, we have discussed the proposed hybrid algorithm CS-FAPP for the path planning. The hybrid algorithm of CS-BAPP is also discussed in this chapter. In both the algorithms, the obstacles are handled with cuckoo search based property with an assumption of considering obstacle as worst nest for cuckoo egg. Path planning is performed using Levy flight search pattern in both the hybrid algorithms with their respective functions.

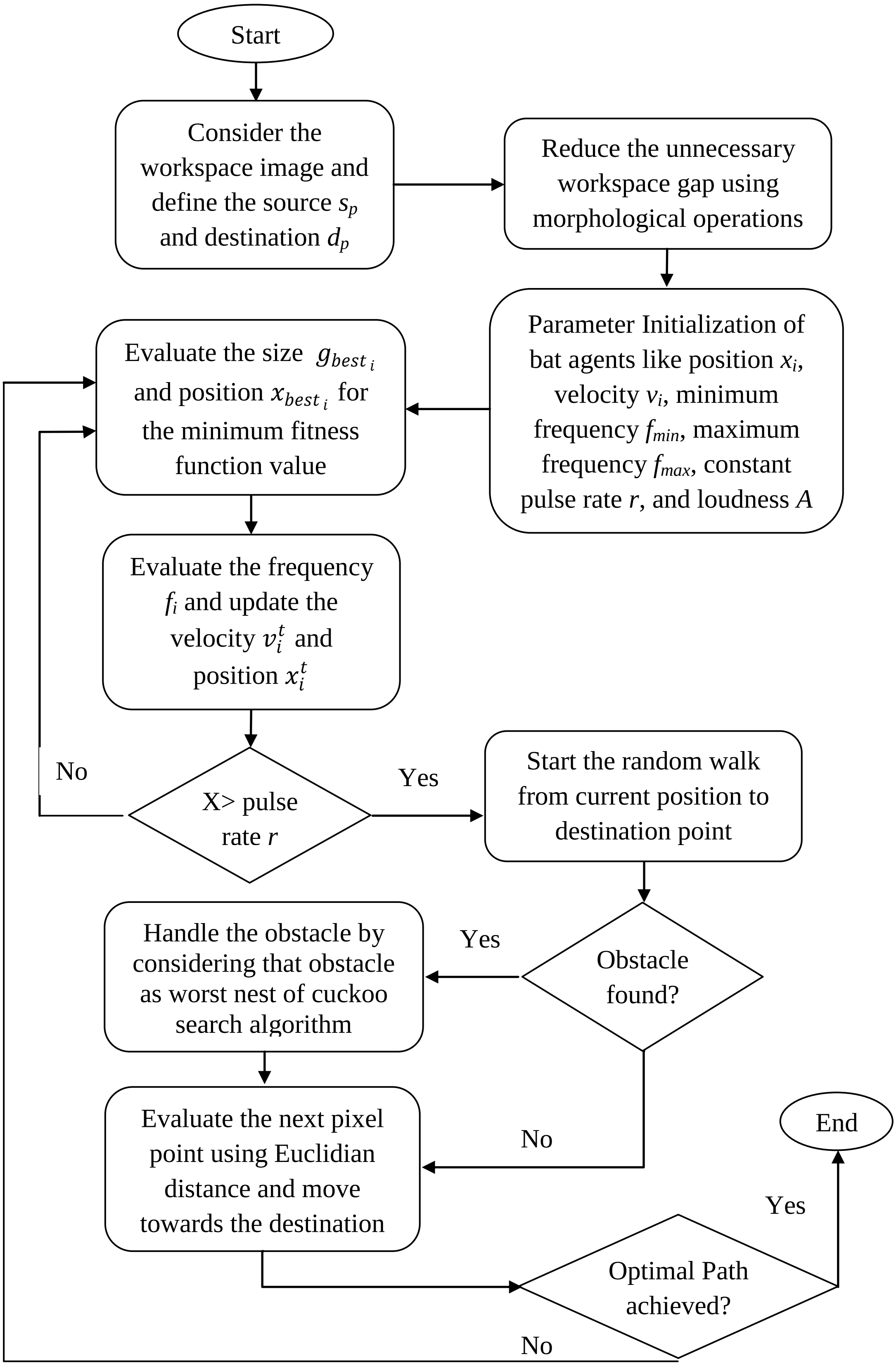


Figure 3.2: Workflow of path planning using hybrid CS-BAPP Approach

Next chapter illustrates the validation of both the hybrid algorithms of CS-FAPP and CS-BAPP based on the standard benchmark functions. Moreover, the experimental setup, information related to MATLAB simulation software, and used database are also discussed. The results and comparison of the proposed CS-FAPP in terms of minimum number of iterations, simulation time, error rate, and success rate are also discussed in the next chapter.

CHAPTER 4
RESULTS AND DISCUSSION

This chapter discusses the computational analysis and validation of the proposed hybrid algorithm of CS-FAPP based on the standard benchmark functions. Also the considered MATLAB simulation tool is discussed in this chapter. Moreover, the results of CS-BAPP are also evaluated and compared as it is the objective of our research work to compare the results of proposed hybrid algorithm of CS-FAPP with hybrid CS-BAPP. Further, the results for the optimal path planning on Google based satellite images are evaluated in terms of evaluation parameters of error rate, success rate, simulation time, and minimum number of iterations,.

4.1. MATLAB SIMULATION SOFTWARE

This section presents the used tool for the simulation of results. Also a brief procedure to generate GUI is elaborated.

There exist a number of tools for the development of the projects. The selection of tools depends upon the user selection and type of work. Here, MATLAB tool has been used for the implementation of the research work. It is a simulation software based tool. The system software of MATLAB has many pre-defined programs. The presence of these programs makes MATLAB tool time effective and easily applicable as well as understandable when compared to the other tools. MATLAB® with its high-level language feature enables programmers to execute tasks with rigorous computations faster than the other programming languages.

MATLAB system is embedded with main five sectional parts:

- **MATLAB Language:** MATLAB is an array or matrix language including many high level language features. The features list of MATLAB consists of object oriented programming features, supports different data structures, functions as well as control statements. It supports “Large Programming type” which supports the generation of large and complex programmes and "Small Programming type" for generation of small as well as quick programs.

- **MATLAB Mathematical Functional Library:** The mathematical functional library of MATLAB is a collection of algorithms to perform computations. This library supports algorithms for small arithmetic operations to the complex Fourier transformations.

- **Working Environment:** The working environment of MATLAB is set of tools and all of the facilities which are necessary for a developer or programmer to work on MATLAB. It includes different application tools considering developing and managing applications, various debugging tools, profiling M-file etc. It also emphasize on data management by providing provisions for data import and export.

- **MATLAB API Library:** The API library of MATLAB is embedded with programming languages like C and FORTAN. Programmers are able to interact with these languages and write the program. The API library also facilitates the user to work with MAT-files.

- **MATLAB Graphic System:** The graphic system of MATLAB is highly advanced as well as user friendly. The user is given excess to perform all graphic functionalities. It supports 2D as well as 3D visualization of the data with numerous graphical features like animations, image processing, rendering etc. MATLAB graphical system also allows user to build a complete Graphical User Interface (GUI) system.

4.1.1. Software Requirements

The software requirements for the MATLAB are listed below:

- Intel Pentium III Processor or Above
- Minimum of 2 GB of RAM

The main window of MATLAB contains the information of Folder currently in use, Command Window, Command History and Workspace as are shown in figure 4.1.

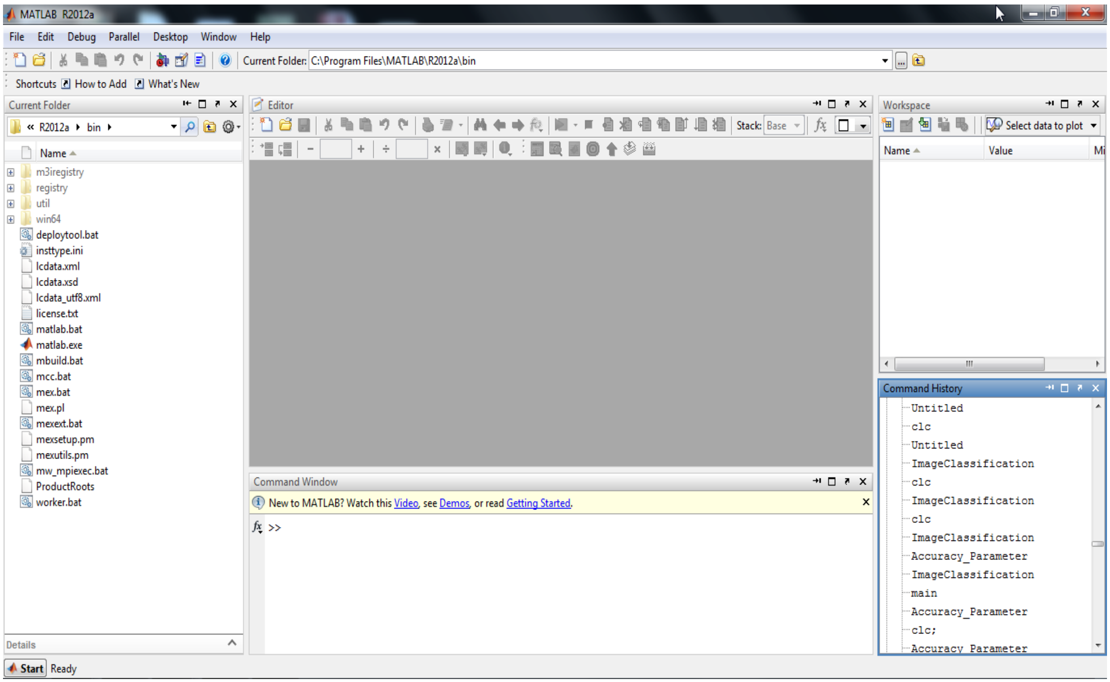


Figure 4.1: MATLAB Simulation Software with Editor, Command Window, Workspace & History

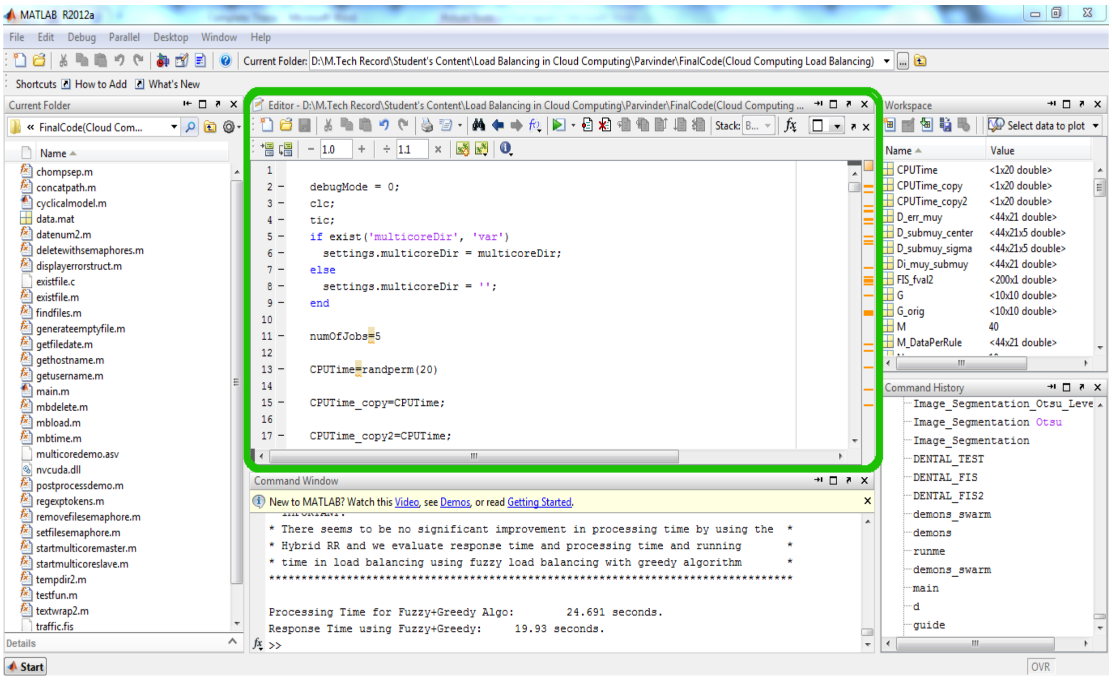


Figure 4.2: MATLAB Editor

All these Workspace, Command History, Editor and Command Window are showing separately in the further snapshots as shown in figure 4.2 to figure 4.5.

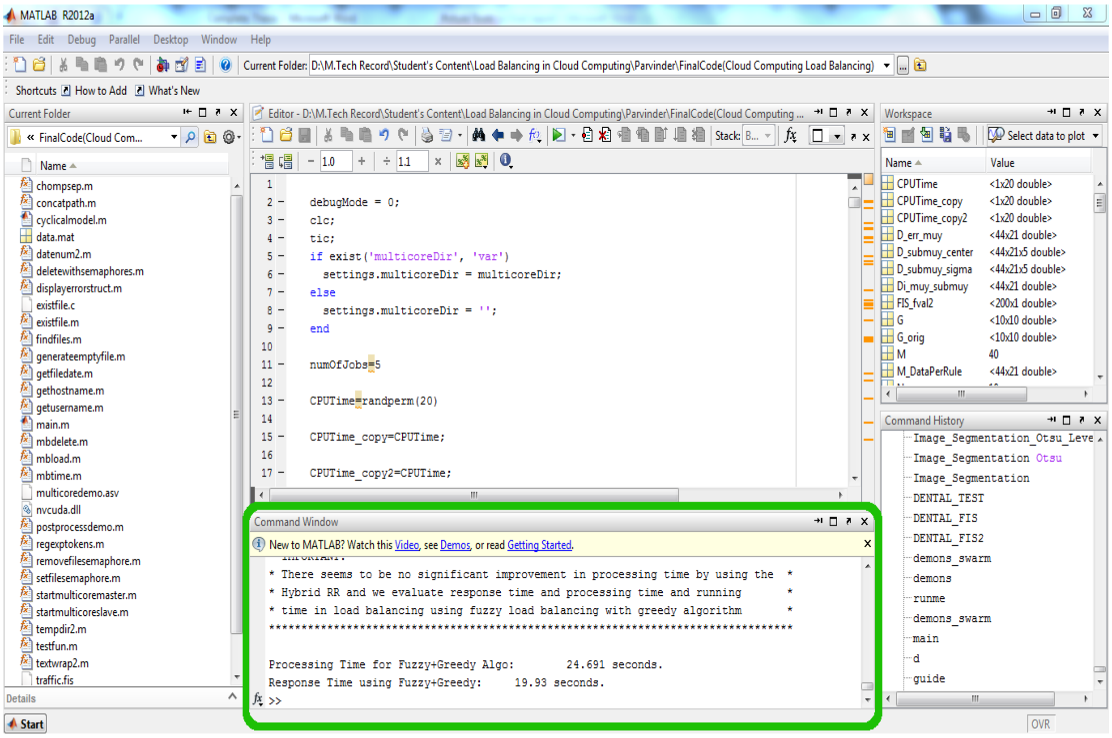


Figure 4.3: MATLAB Command Window

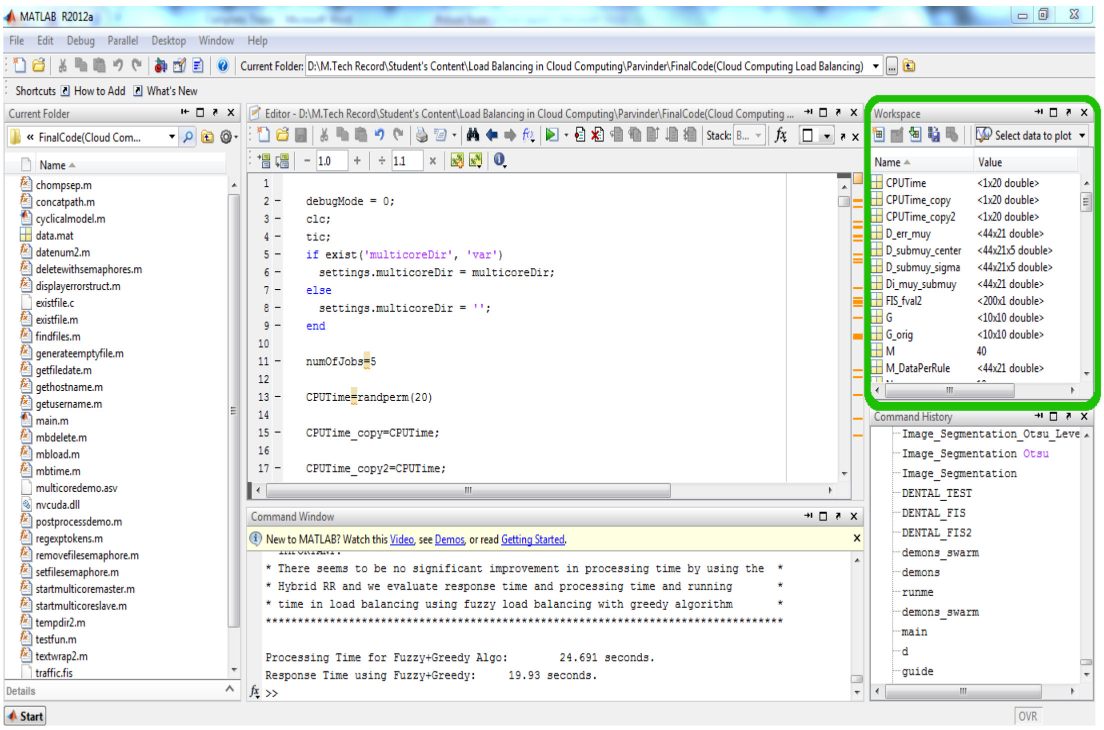


Figure 4.4: MATLAB Workspace

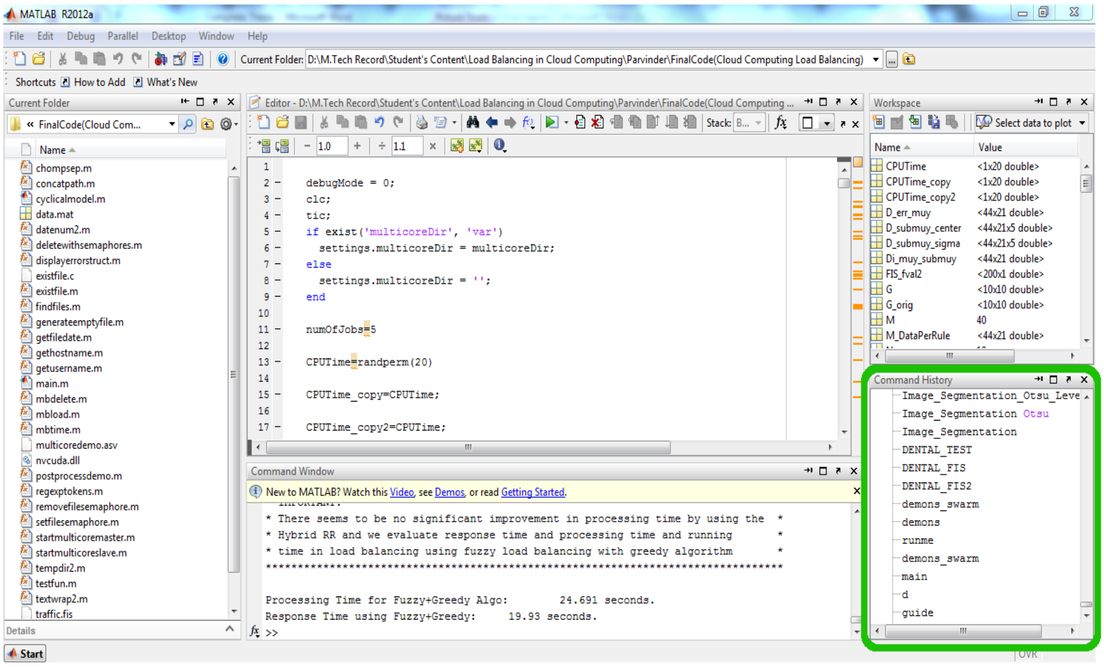


Figure 4.5: MATLAB Command History

4.1.2. GUI

Figure 4.6 illustrates that how to make a novel GUI:-

- To construct a novel GUI, go to, File< New< GUI as illustrated in fig. 4.6.

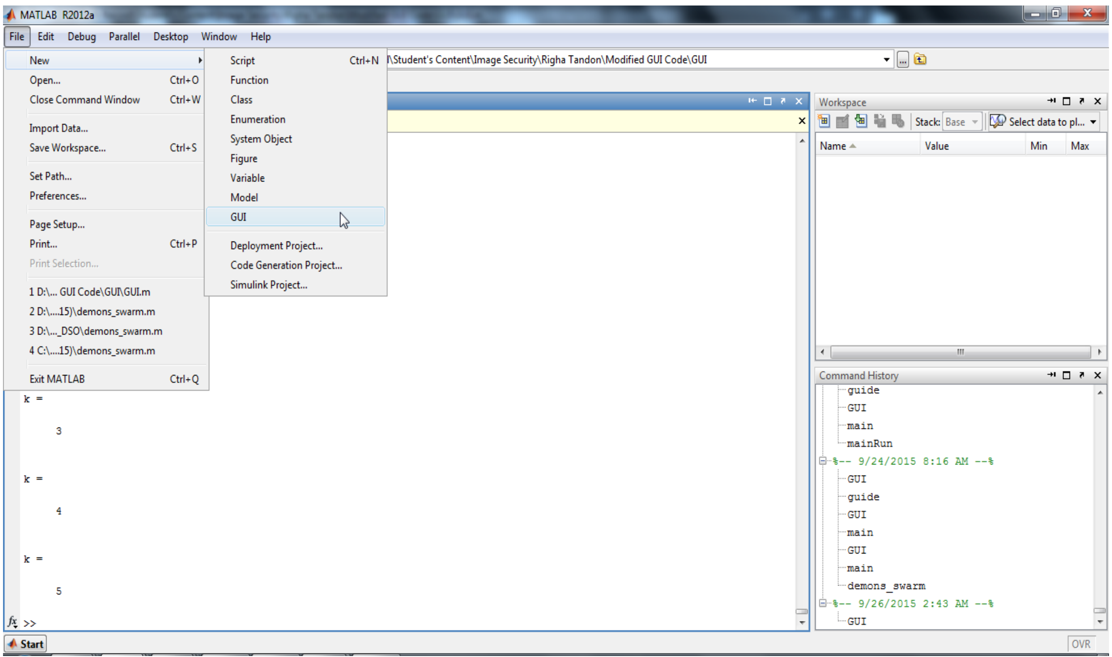


Figure 4.6: New GUI

- Figure 4.7 presents the way to create a novel GUI or to open existing one.
- Blank GUI’s can be made or GUI’s with some options.
- Figure 4.8 displays the platform to draw GUI.

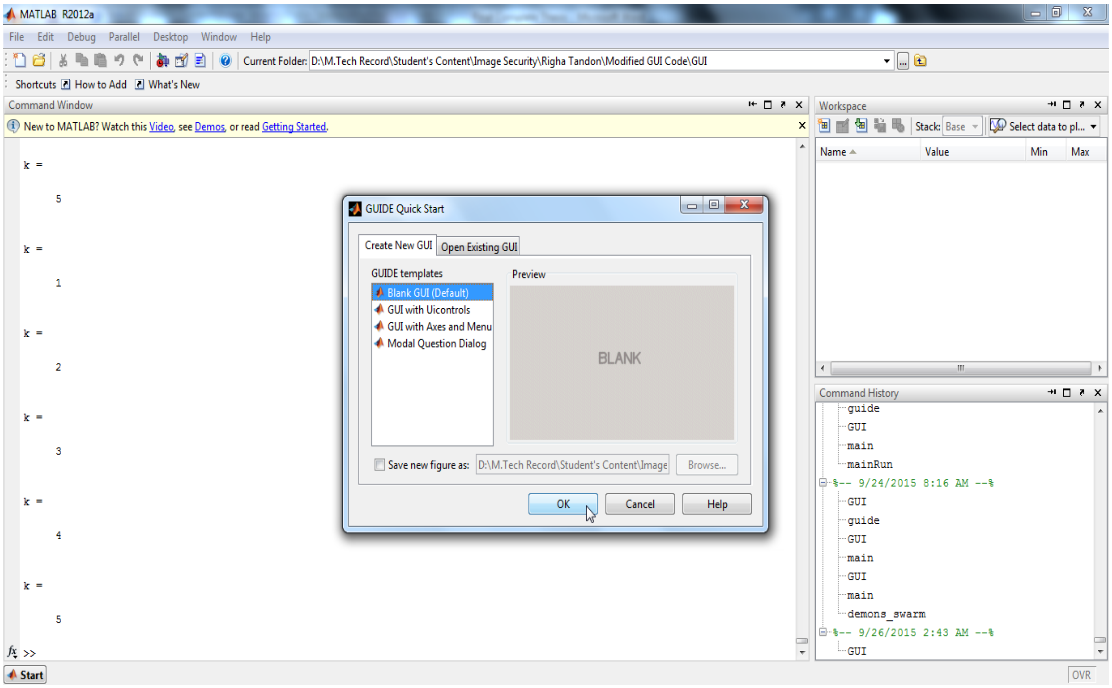


Figure 4.7: Other Options for GUI

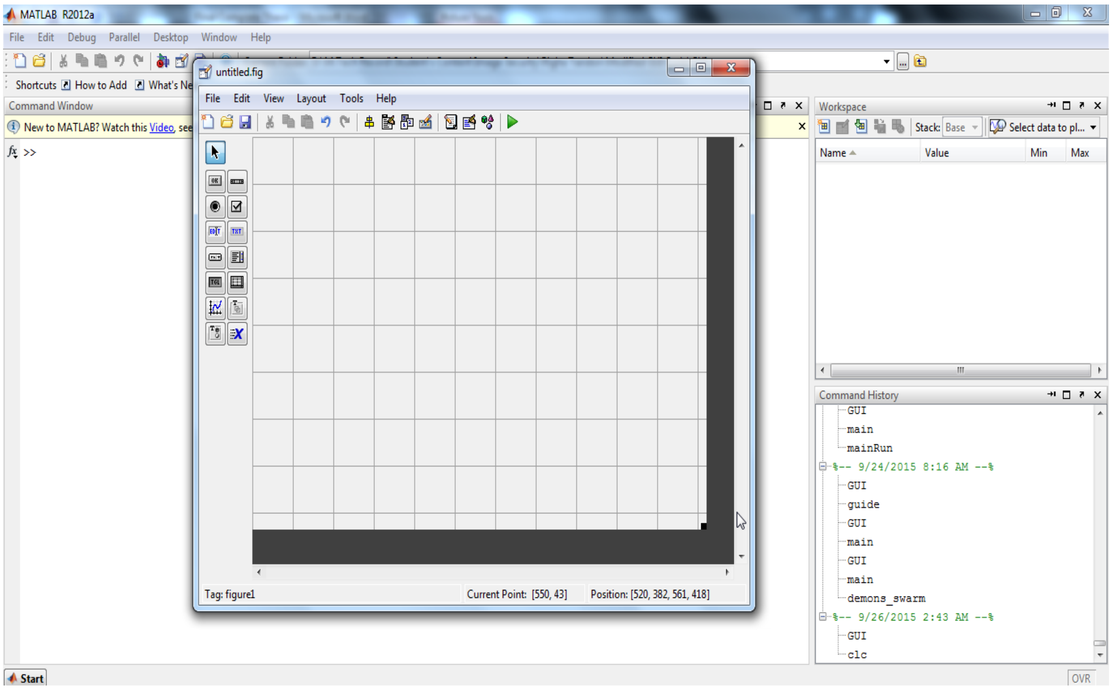


Figure 4.8: To Draw a GUI

4.2. BENCHMARK FUNCTION TESTING

Before applying the proposed hybrid algorithm of CS-FAPP for any real time routing applications, it is essential to analyze the efficacy of the proposed hybrid systems with benchmark functions. There are a lot of benchmark functions to test the efficacy of algorithms based on shape and physical properties. In this research work, we have considered the benchmark functions of Rosenbrock, Michalewicz, Ackley, Easom, De Jong, Schwefel, Rastrigin, Griewank, and Shubert function. The primary objective for the selection of mentioned benchmark functions is the availability of results values for these functions using individual FA [183], BA [184] and CS [185]. The evaluated results of hybrid CS-FAPP and CS-BAPP based on mentioned benchmarks functions have been performed on window 7 based computer system having confirmation of Intel i5 Processor and 4GB of RAM. Results are recorded using MATLAB simulation software version 8.3.0.532 and compared with results values evaluated with individual FA [183], BA [184] and CS [185]. The considered benchmark functions are discussed here.

4.2.1. Ackley Function

It is an n-dimensional non convex function proposed by David Ackley in 1987 [186]. It possesses multiple local minima and single global minima with value *0* at *x=0*. It can be evaluated using Equation (4.1).

$$f(x) = -a\exp\left(-b\sqrt{\frac{1}{d}\sum_{i=1}^{d} x_i^2}\right) - \exp\left(\frac{1}{d}\sum_{i=1}^{d}\cos(cx_i)\right) + a + \exp(1)$$

… Equation (4.1)

Where, a, b, & c are the constant values, d stands for dimensions and $x_i \in [-32,32]$ for all $i = 1,2,3,\ldots d$. In this research work, we have used $a = 20$, $b = 0.2$, $c = 2\pi$, and $d = 128$.

4.2.2. Shubert Function

Shubert function is continuous n-dimensional non-convex function. It possesses various local minima and multiple global minima features. It can be evaluated using equation (4.2).

$$f(x) = \left(\sum_{i=1}^{5} i\cos\big((i+1)x_1 + i\big)\right)\left(\sum_{i=1}^{5} i\cos\big((i+1)x_2 + i\big)\right)$$

… Equation (4.2)

Where, the any input can be used to define the function but mostly considerable domain is $x_i \in [-10,10]$ for *i = 1, 2*. Function is evaluated with global minima value of d =18.

4.2.3. Griewank Function

Griewant function is also continuous unimodal that can be defined for the n-dimensional space. Griewant function defines the convergence of optimization function. It can be evaluated using equation (4.3). The function can be defined for the hypercube based input values of $x_i \in [-600,600]$ for *i = 1,2, …, d*.

$$f(x) = \sum_{i=1}^{d} \frac{x_i^2}{4000} - \prod_{i=1}^{d} \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$$

… Equation (4.3)

4.2.4. Rastrigin Function

Rastrigin Function is convex, multimodal and continuous function n-dimensional function. It possesses local minima but the locations of local minima are always distributed. It can be evaluated using equation (4.4). the function can be defined for the hypercube input values of $x_i \in$ *[-5.12,5.12]* for *i = 1,2, …, d*.

$$f(x) = 10d + \sum_{i=1}^{d} [x_i^2 - 10\cos(2\pi x_i)]$$

… Equation (4.4)

4.2.5. Schwefel Function

It is non-separable, non-differentiable, non-convex, multimodal function. It also possesses several local minima and can be evaluated for the input $x_i \in [-500{,}500]$, where *i = 1,2, …, d*. Here, d is considered 128. The function can be evaluated using equation (4.5)

$$f(x) = 418.9829d - \sum_{i=1}^{d} x_i \sin\left(\sqrt{|x_i|}\right)$$

… Equation (4.5)

4.2.6. De Jong Function

De Jong function is sharp drop function on flat surface. It possesses the property of mutimodal and can be evaluated for the input x_i ∈ *[-65.536, 65.536]* for *i = 1,2*. Here, the function is evalauted using equation (4.6) for d = 256.

$$f(x) = \left(0.002 + \sum_{i=1}^{25} \frac{1}{i + (x_1 - a_{1i})^6 + (x_2 - a_{2i})^6}\right)^{-1}$$

… Equation (4.6)

Where,

$$a = \begin{pmatrix} -32 & -16 & 0 & 16 & 32 & -32 & \dots & 0 & 16 & 32 \\ -32 & -32 & -32 & -32 & -32 & -16 & \dots & 32 & 32 & 32 \end{pmatrix}$$

4.2.7. Easom Function

Easom function is non-scalable, non separable continuous function that has multiple local minima but limited global minima as compared to available search space. It can be evaluated using equation (4.7) for the input x_i ∈ *[-100,100]* for all *i = 1,2*.

$$f(x) = -\cos(x_1)\cos(x_2)\exp(-(x_1 - \pi)^2 - (x_2 - \pi)^2)$$

… Equation (4.7)

4.2.8. Michalewicz Function

It is multimodal function with several local minima. It is used to estimate the ridges and valleys. It can be evaluated using equation (4.8) for the input search space of $x_i \in [0, \pi]$ for *i = 1,2,…,d.*

$$f(x) = -\sum_{i=1}^{d} \sin(x_i) \sin\left[\frac{ix_i^2}{\pi}\right]^{2m}$$

… Equation (4.8)

Where, *m* is the steepness of search space. The higher value of *m* makes it difficult for the search function. This function is evaluated for *d = 16*.

4.2.9. Rosenbrock Function

This function is Unimodal function used to estimate the valley information. It possesses very narrow and limited global minima search space value. It can be evaluated for the input $x_i \in$ *[-5, 10]* but restricted to values of $x_i \in$ *[-2.048, 2.048]* for all *i = 1,2,…, d.* It can be evaluated using equation (4.9). In this research work, value of *d* is considered as 16.

$$f(x) = \sum_{i=1}^{d-1} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$$

… Equation (4.9)

Based on the above mentioned equations (4.1)-(4.9), benchmark functions are evaluated. The results are evaluated using MATLAB software with maximum possible repetition of hybrid approach to better analyze the results. The global optima results are evaluated until variation of results is observed lesser than 10^{-5}. The evaluated values of these benchmarks functions using hybrid proposed algorithms along with values of individual FA, BA and CS are illustrated in table 4.1. In table 4.1, values are expressed as average number of evaluations in success rate which is presented in the format of $x \pm y$, where *x* is the mean of function evaluation value and *y* is the standard deviation.

Table 4.1: Comparison of proposed hybrid CS-FAPP with CS-BAPP and individual FA, CS, & BA based on Benchmark Functions

Benchmark Function	CS	BA	FA	CS-BAPP	CS-FAPP
Ackley Function (d=128)	4936 ± 903	6933 ± 2317	4392 ± 2710	4611 ± 1354	3912 ± 1207
Shubert Function (18 minima)	9770 ± 3592	11925 ± 4049	9925 ± 2504	9561 ± 3108	9242 ± 2098
Griewank Function	10912 ± 4050	9792 ± 4732	10790 ± 2977	9432 ± 3991	9163 ± 3103
Rastrigin Function	10354 ± 3755	12573 ± 3372	12075 ± 3750	9718 ± 3527	8917 ± 3079
Schwefel Function (d=128)	8829 ± 625	8929 ± 729	7923 ± 524	7866 ± 693	6341 ± 540
De Jong Function (d=256)	4971 ± 754	5273 ± 490	5657 ± 730	4715 ± 416	4492 ± 521
Easom Function	6751 ± 1902	7532 ± 1702	6082 ± 1690	6243 ± 2133	5152 ± 1667
Michalewicz Function (d=16)	3221 ± 519	4752 ± 753	2889 ± 719	2856 ± 632	2371 ± 487
Rosenbrock Function (d=16)	5923 ± 1937	7923 ± 3293	6040 ± 535	5189 ± 1458	4583 ± 626

From the performance evaluation results presented in table 4.1, it can be analyzed that proposed hybrid concept CS-FAPP is superior in comparison with CS-BAPP and individual BA, FA, and CS algorithm. Results are achieved with optima global minima with comparative lesser standard deviation of results. Moreover, performance evaluation of hybrid CS-FAPP algorithm also indicates the maintainability of balance between the exploration and exploitation. These evaluations make of hybrid concepts made confident to use for the optimal path planning on real time application areas of India. Before evaluating the results for real time application (satellite images) regions of India, path planning using proposed hybrid CS-FAPP algorithm is discussed for the working example of a small manually defined 8*8 matrix region.

4.3. IMPLEMENTATION ON HUMAN ANNOTATED WORKSPACE

This section explained the proposed hybrid algorithm with the help of an example. The objective is to find out the shortest path from specified source and destination point without any collision. The workspace environment is static and unknown in nature i.e. there is no prior information of the obstacles. Figure 4.9 presents the workspace area with multiple obstacles to explain the working of proposed hybrid algorithm for optimal path planning.

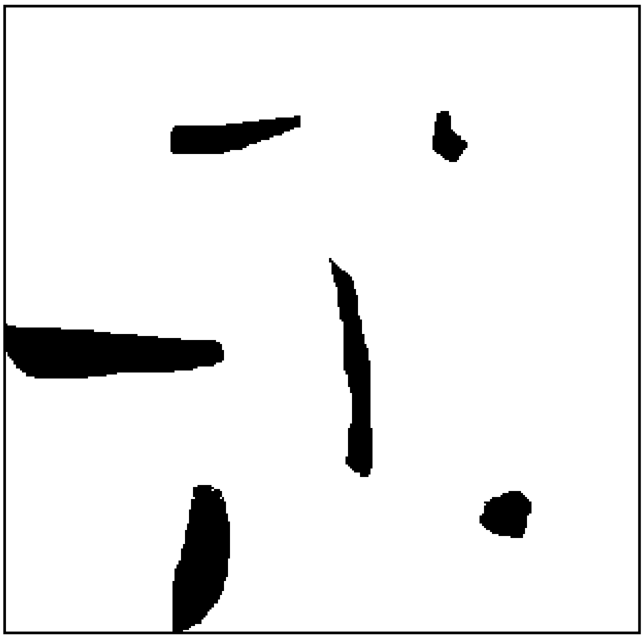


Figure 4.9: Workspace with multiple obstacles

The workspace is considered in the form of pixel grid with black and white colours. The pixel point with black colour indicates the presence of obstacle and white colour indicates pixel point without any obstacle as presented in figure 4.10.

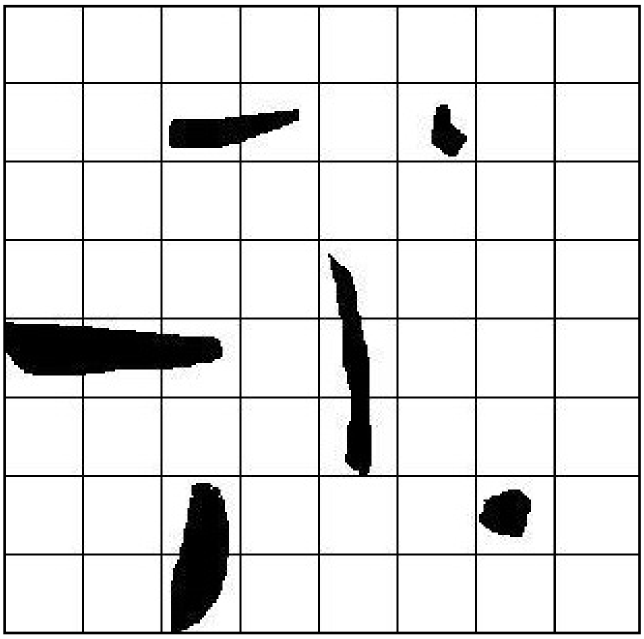


Figure 4.10: Workspace Division in the form of Pixel Grid

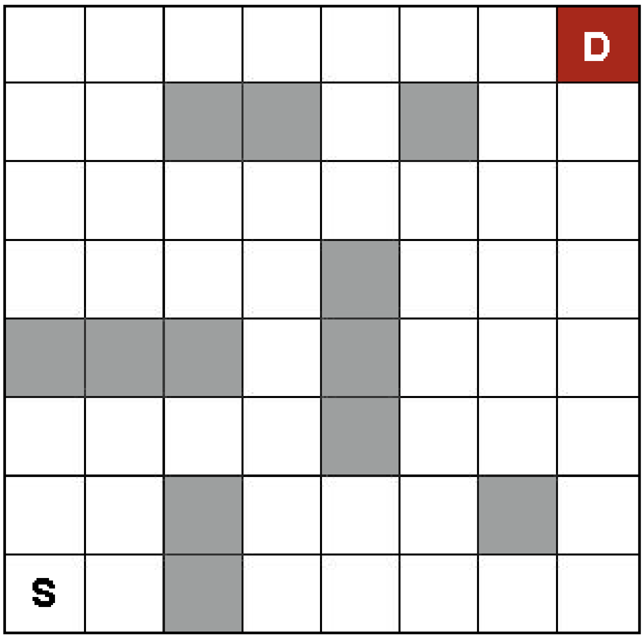


Figure 4.11: Filled Obstacle Region of Workspace

The region of working space which contains the obstacle area is complete filled as a black region of the grid which is presented in figure 4.11. An 8*8 pixel grid is considered with multiple obstacles and is numbered from 1 to 64 where number 1 is the source point (S) and number 64 is the destination point (D) as presented in figure

4.12. The pixel points: 3,11,15,21,25,26,27,29,37,51,52 and 54 are the obstacle points. In case if a pixel point is partially covered by obstacle i.e. black colour then the whole pixel point is considered as black pixel (pixel point with obstacle).

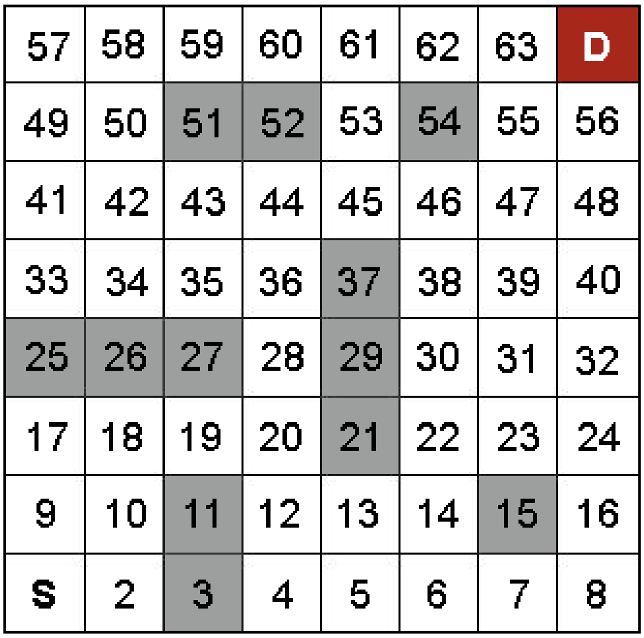


Figure 4.12: Workspace Grid with Specified Source and Destination Point

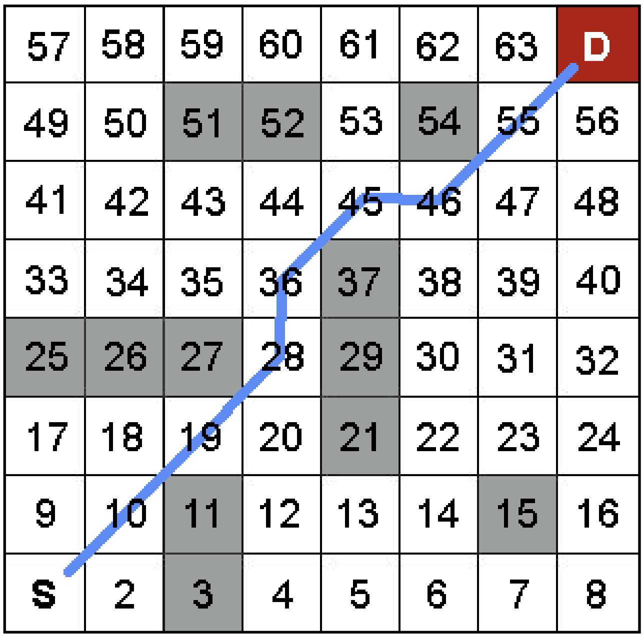


Figure 4.13: The best path from S to D

Due to the attractiveness property of fireflies i.e. one firefly is attracted towards brighter firefly, the brightest firefly is placed at destination D. The destination pixel point is represented in red colour and source point is indicated with alphabet S. In the proposed algorithm it is assumed that *n* number of fireflies contains *n* number of candidate solution and one amongst them is the optimal solution. The firefly population is initialized starting from source point and one firefly which is the brightest is placed at destination point in order to attract other fireflies. The fireflies use the pattern mimicking property of cuckoo bird to detect and handle obstacles. Here when a black pixel is encountered by a firefly, it means an obstacle is detected and then the Euclidian distance of the neighbour pixel points from the destination point is calculated. The pixel point with minimum distance value is considered as next pixel point for further processing. Figure 4.13 presents the best/optimum collision free path from source to destination point. The path is S-10-19-28-36-45-46-55-D.

Further the experimentation is performed based on the real time Google based Satellite images of different regions of India. The detailed description of these satellite images is discussed further.

4.4. IMPLEMENTATION ON GOOGLE BASED SATELLITE IMAGES

For experimentation, satellite images of different regions of India are extracted using the Google Map. The extracted images are of different regions, terrains and size. The images consist of terrain features of water, vegetation, hilly (rocky), urban, and barren with major interest on the vegetation feature. The considered satellite images are illustrated in figures 4.14 (a)-(e). All the images are of different size and taken from different regions representing the terrain features with diversity. The satellite image 1 of size 267*265 pixels consists of mainly the vegetation region along with barren and urbanization terrains and captured from the nearby region of Mahodra, Rajasthan, India. The satellite image 2 of size 307*240 pixels also contains the terrain features of vegetation, urbanization and barren. This image is taken from the Kelwara region of Rajasthan, India. The satellite image 3 of size 203*258 is taken from the Tulera

Region, Alwar (Rajasthan), India. This image consists of mainly the terrain feature of dense vegetation along with some urbanization sector. The next satellite image 4 of size 331*248 is taken from the border line regions of Rajasthan and Madhya Pradesh states of India. The nearby regions are Ghatta region of Rajasthan and Parwah region of Madhya Pradesh, India. This image consists of terrain features of water, vegetation, barren, and urbanization. The last satellite image 5 of size 405*316 is taken from the nearby Hedri, Maharashtra, India which consists of mainly the terrain regions of hilly and vegetation. The brief information of the considered satellite images is also mentioned in table 4.2.

(a) Satellite Image 1

(b) Satellite Image 2

(c) Satellite Image 3

(d) Satellite Image 4

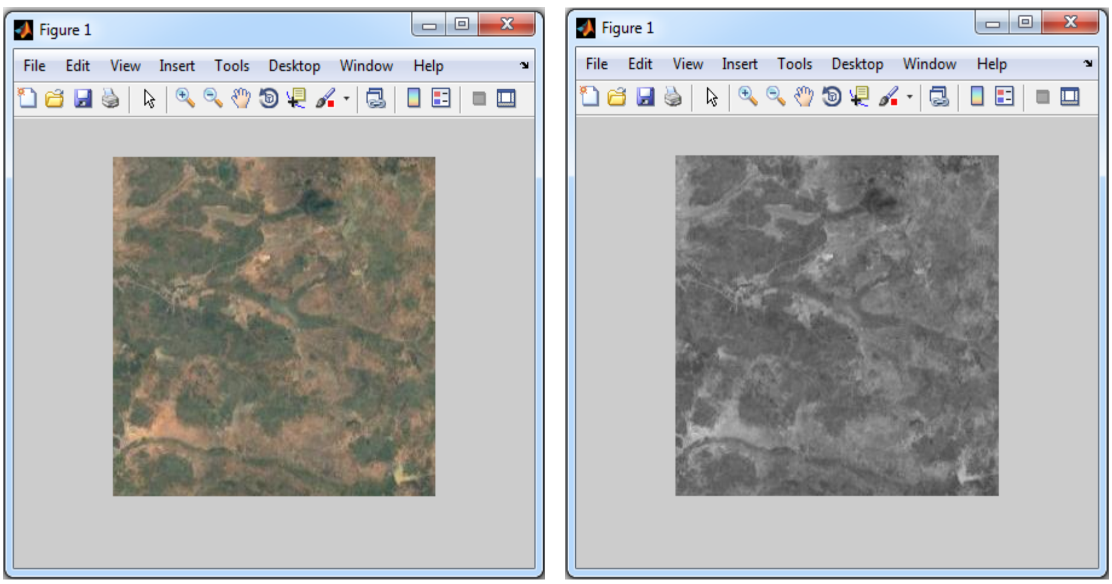
(e) Satellite Image 5

Figure 4.14 (a)-(e): Google based Satellite Images considered for Experimentation

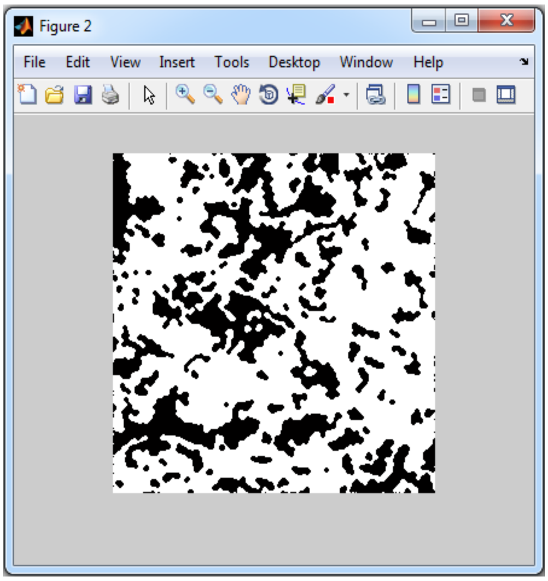
Table 4.2: Description of Considered Satellite Images for Experimentation

Image	Size	Region	Terrain Features
Satellite Image 1	267*265	Mahodra, Rajasthan	Vegetation, Barren, and Urban
Satellite Image 2	307*240	Kelwara, Rajasthan	Vegetation, Barren, and Urban
Satellite Image 3	203*258	Tulera Region, Alwar, Rajasthan	Vegetation and Urban
Satellite Image 4	331*248	Ghatta, Rajasthan and Parwah, Madhya Pradesh, India	Vegetation, Water, Barren, and Urban
Satellite Image 5	405*316	Hedri, Maharashtra	Vegetation and Hilly (Rocky)

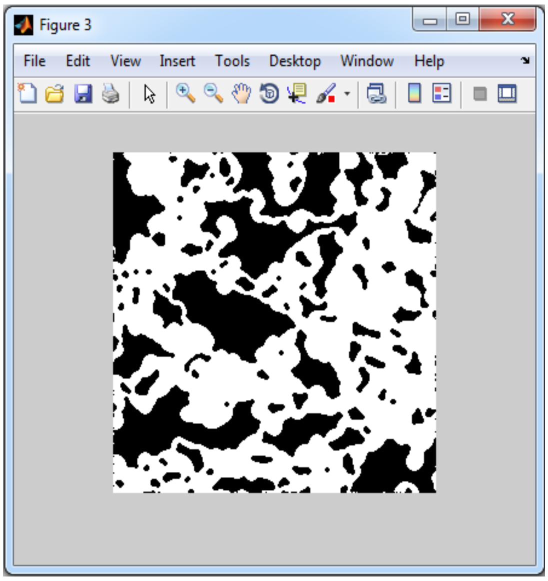
The application of path planning using the proposed CS-FAPP algorithm is performed on mentioned satellite images. The results of path planning are illustrated in five images of consideration of satellite image, conversion into grey scale image, preprocessing and conversion into binary image, morphological processing, and final obtained optimal path. As mentioned, initially the satellite image is considered, then the considered satellite image is converted to grey scale image as the images taken from the Google map contains different color combination representing different land cover attributes. Then, the image is pre-processed and converted into binary form. This binary image workspace is further considered as collection of binary pixels values: 0 and 1 where value 1 indicates obstacle free white pixel on which social agents can move and value 0 is black pixel that indicates the presence of obstacle. In this binary image, small noise and smaller obstacles can be observed. Some smaller gaps are within the region of bigger boundary area where social agents can't enter to find the path due to closure of boundary regions. To overcome these useless inner closed regions, morphological operation is applied and the small neighboring obstacles are transformed into one bigger obstacle. It also reduces the complexity of algorithm. The image obtained after the morphological operations is considered for the final path planning. The final obtained optimal path is stored along with evaluated path length. This path planning process on all the satellite images is illustrated in figure 4.15-4.19 for the satellite image 1-5 respectively.



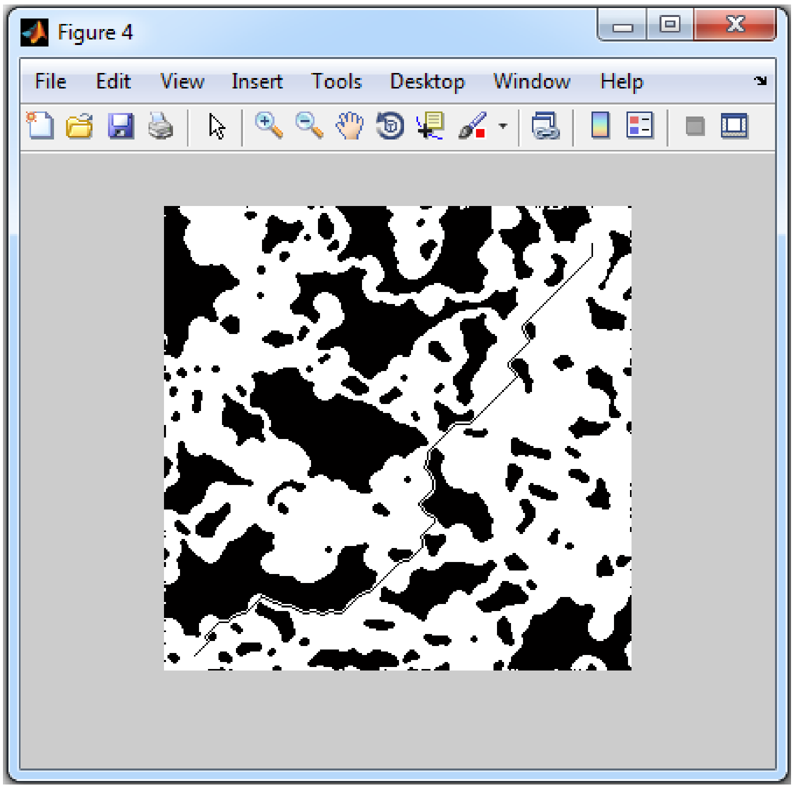
(a) Consideration of Satellite Image 1 **(b) Conversion to Grey Scale Image**



(c) Conversion to binary image



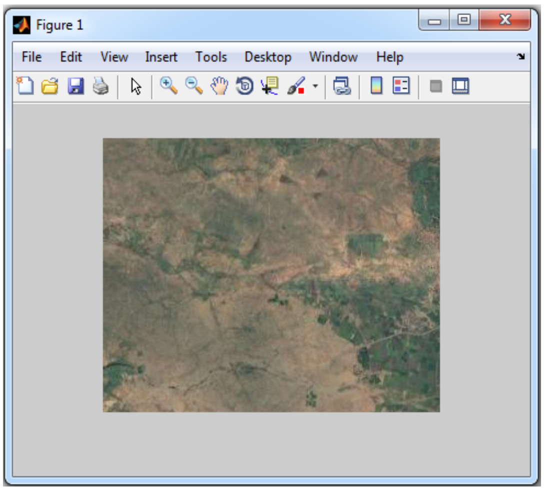
(d) Morphologically Operated Image



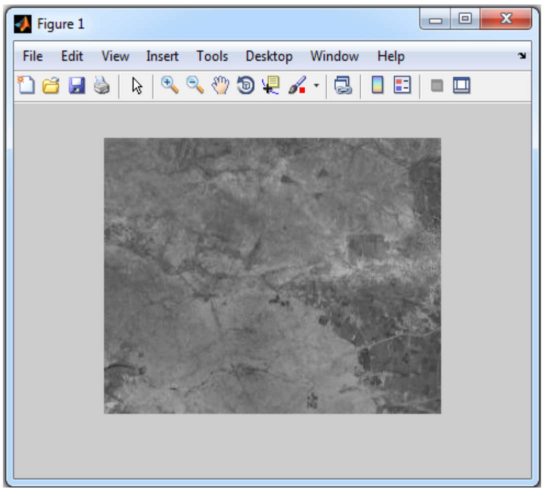
(e) Obtained Optimal Path of Satellite Image 1

Figure 4.15 (a)-(e): Path Planning Results of Satellite Image 1

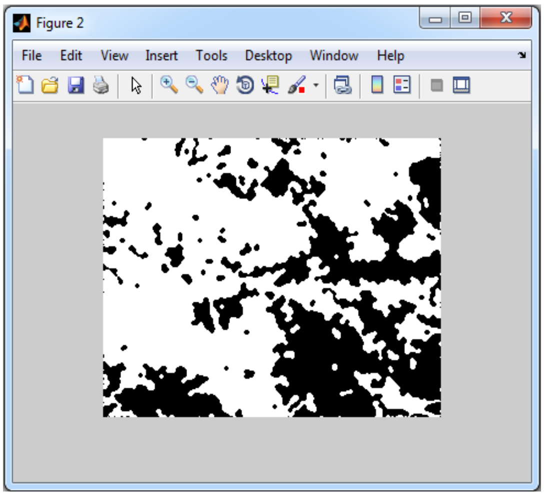
The figure 4.15 (a)-(e) illustrated the process of optimal path planning using proposed CS-FAPP approach for satellite image 1. The size of satellite image 1 is 267*265 pixels. For path planning, the defined start point is (18, 257) and destination point is (245, 22). On the basis of defined start and destination points, the proposed CS-FAPP approach has obtained the optimal path length of 320 pixels. Further, the results for the satellite image 2 are evaluated.



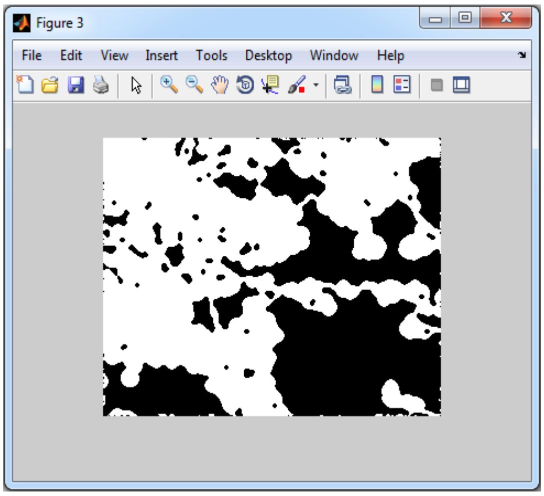
(a) Consideration of Satellite Image 2



(b) Conversion to Grey Scale Image

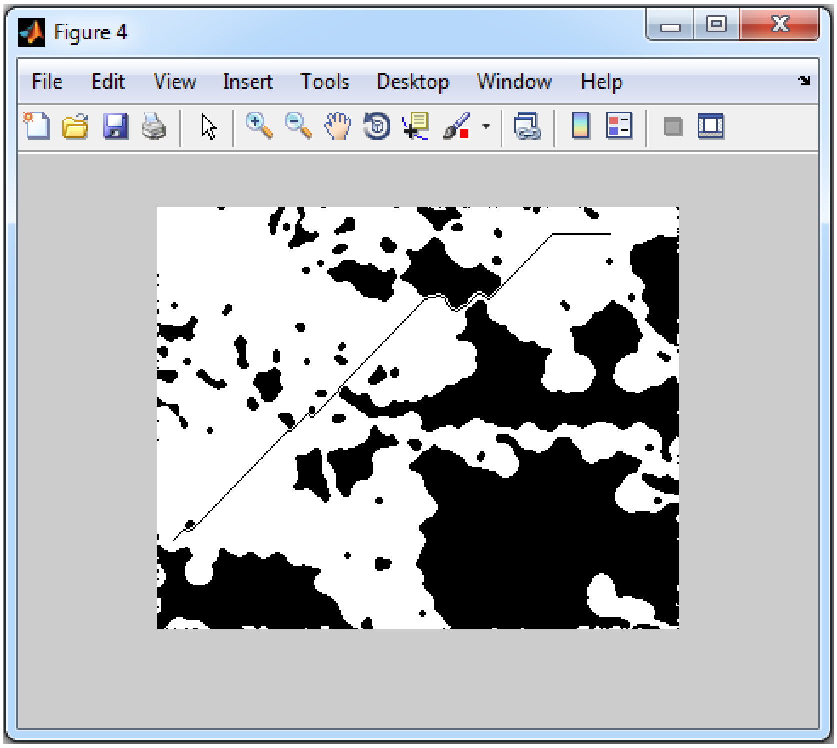


(c) Conversion to binary image



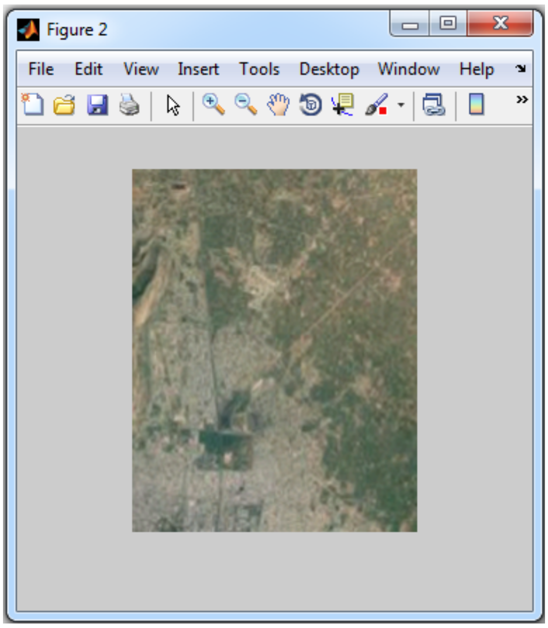
(d) Morphologically Operated Image

The process of optimal path planning using CS-FAPP is also successfully accomplished by satellite image 2 as illustrated in figure 4.16 (a)-(e). The size of satellite image 2 is 307*240 pixels. For path planning, the defined start point is (10, 190) and destination point is (267, 16). On the basis of defined start and destination points, the proposed CS-FAPP approach has obtained the optimal path length of 261 pixels as illustrated in figure 4.16 (e). Further, the results for the satellite image 3 are evaluated.

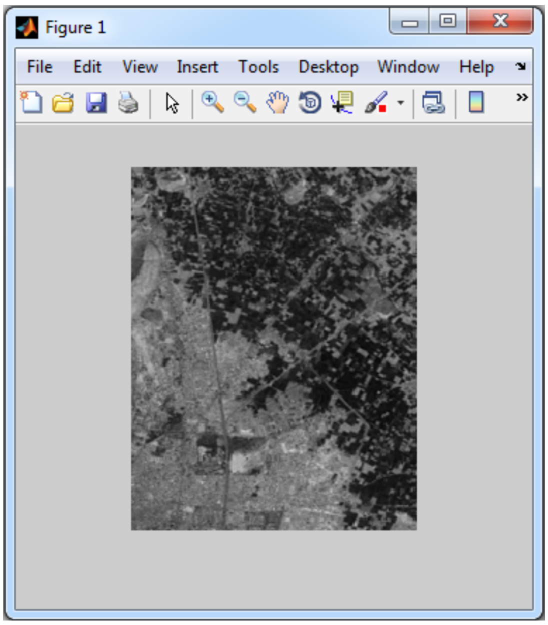


(e) Obtained Optimal Path of Satellite Image 2

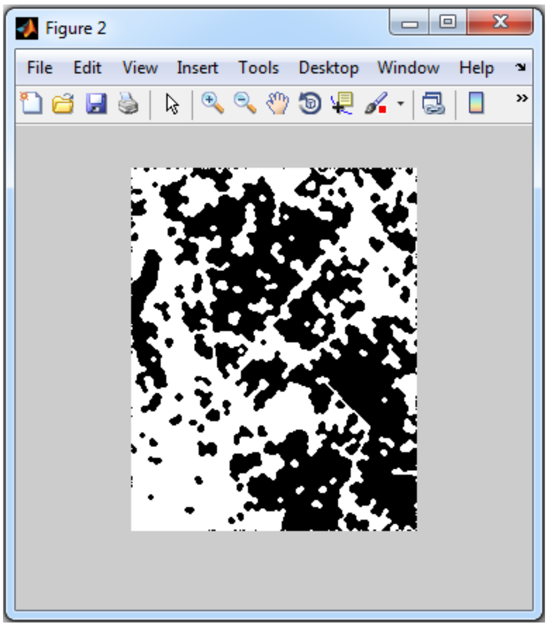
Figure 4.16 (a)-(e): Path Planning Results of Satellite Image 2



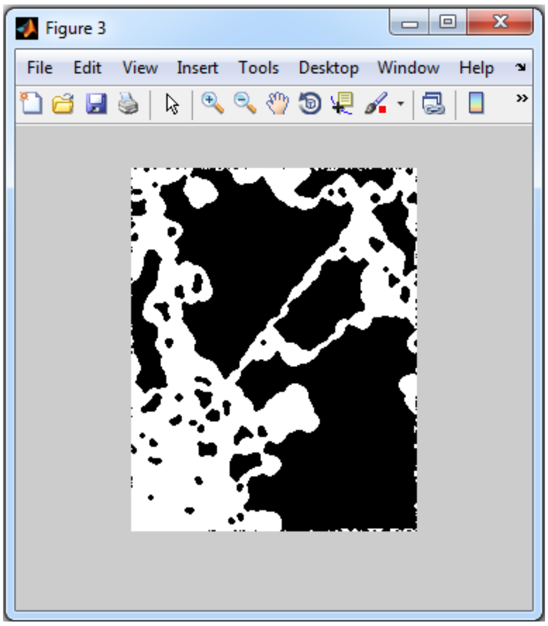
(a) Consideration of Satellite Image 3



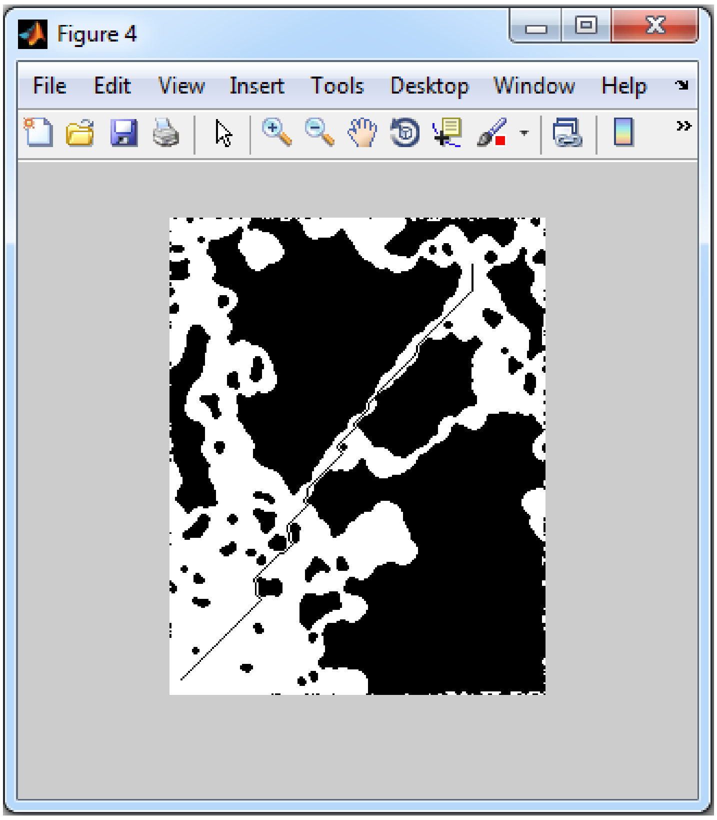
(b) Conversion to Grey Scale Image



(c) Conversion to binary image



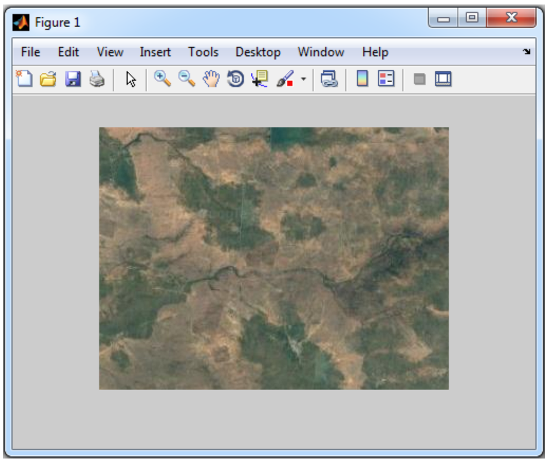
(d) Morphologically Operated Image



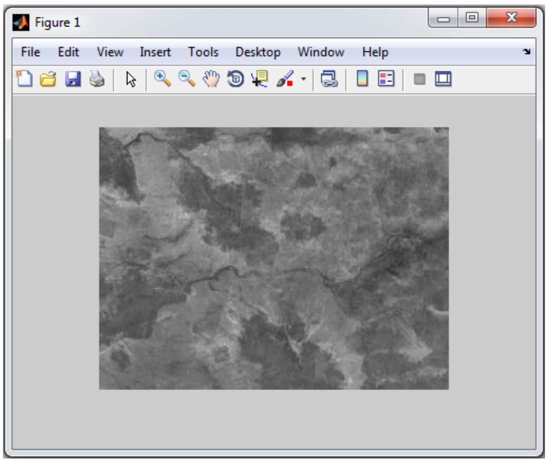
(e) Obtained Optimal Path of Satellite Image 3

Figure 4.17 (a)-(e): Path Planning Results of Satellite Image 3

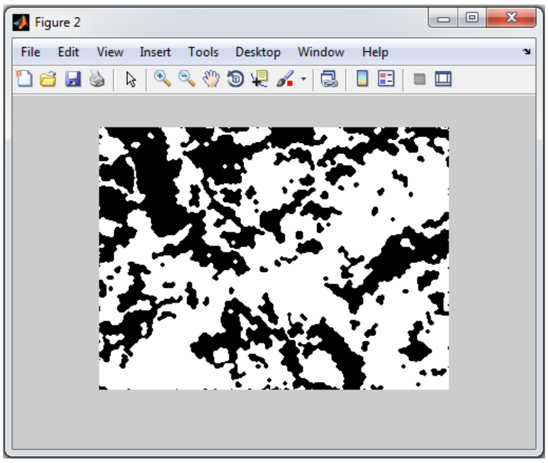
Further, the figure 4.17 (a)-(e) illustrates the path planning process using proposed CS-FAPP approach for the satellite image 3. The size of satellite image 3 is 203*258 pixels. For path planning, the defined source and destination points for satellite image 3 are (7, 250) and (164, 26) respectively. On the basis of defined initial source and destination points, the proposed CS-FAPP approach has obtained the optimal path length of 246 pixels as illustrated in figure 4.17 (e). Further, the results for the satellite image 4 are evaluated in figure 4.18 (a)-(e).



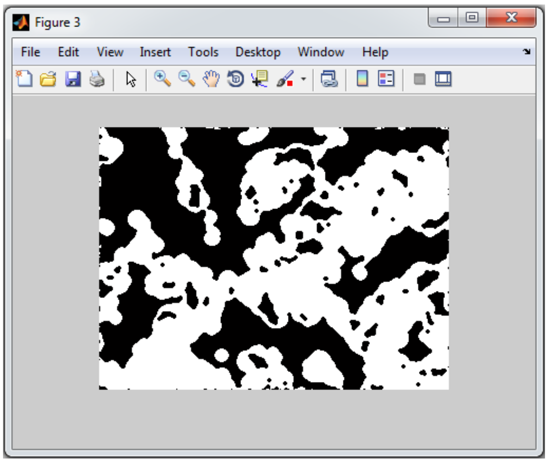
(a) Consideration of Satellite Image 4



(b) Conversion to Grey Scale Image

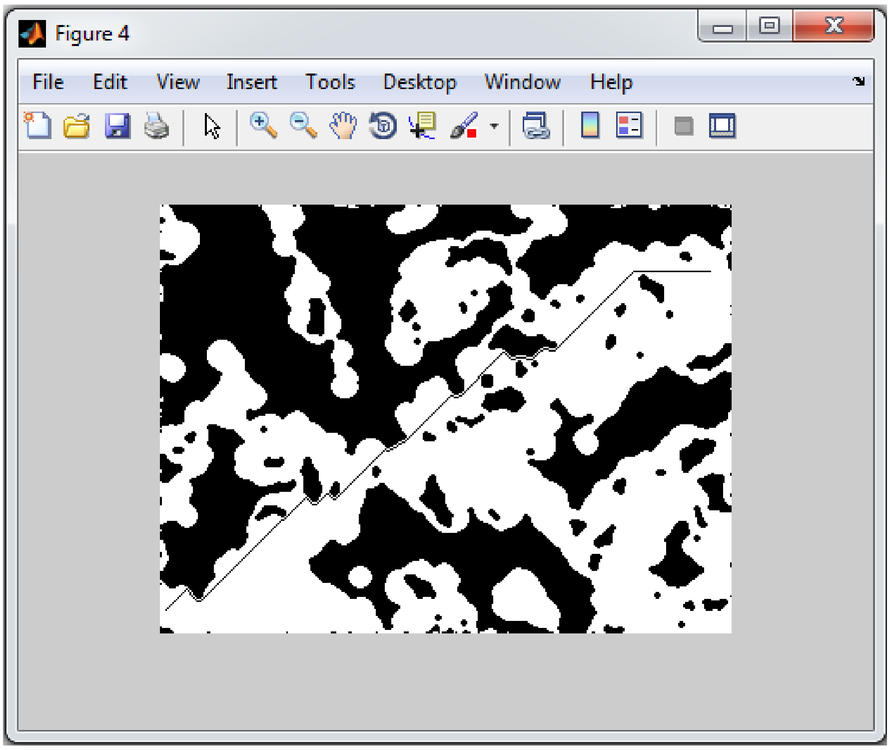


(c) Conversion to binary image



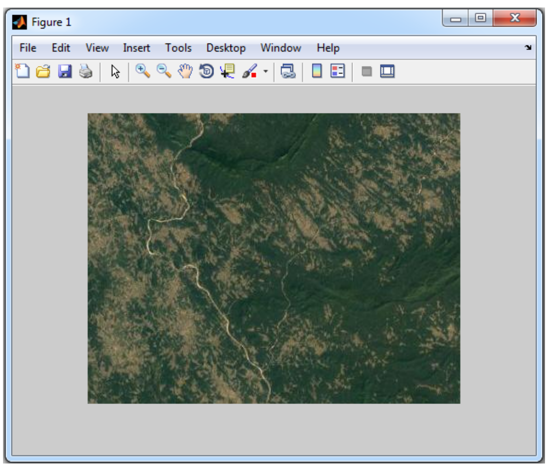
(d) Morphologically Operated Image

Further, the figure 4.18 (a)-(e) illustrates the path planning process for the satellite image 4 having size of is 331*248 pixels. The defined initial source and destination target points for satellite image 4 are (4, 235) and (319, 39) respectively. On the basis of defined initial source and destination points, the proposed CS-FAPP approach has obtained the optimal path length of 318 pixels as illustrated in figure 4.18 (e). Further, the results for the satellite image 5 are evaluated in figure 4.19 (a)-(e).

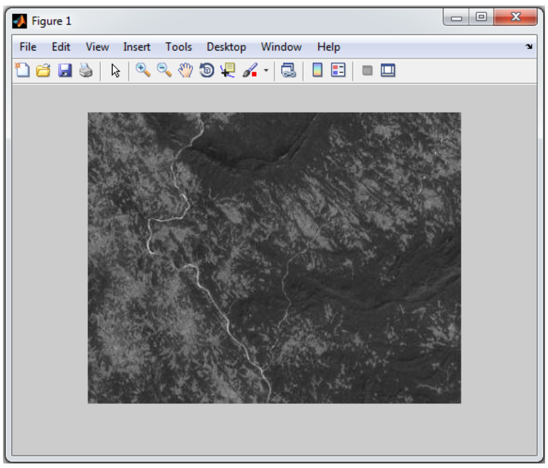


(e) Obtained Optimal Path of Satellite Image 4

Figure 4.18 (a)-(e): Path Planning Results of Satellite Image 4

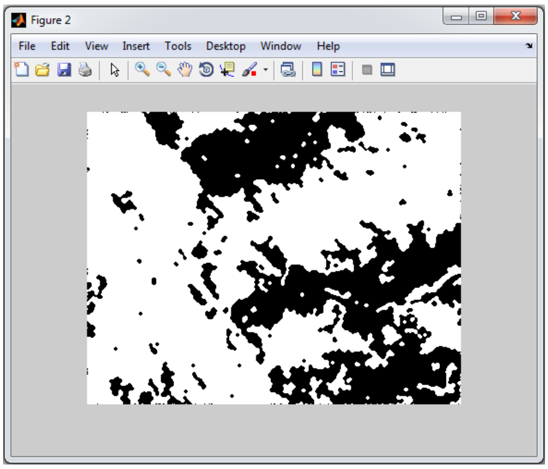


(a) Consideration of Satellite Image 5

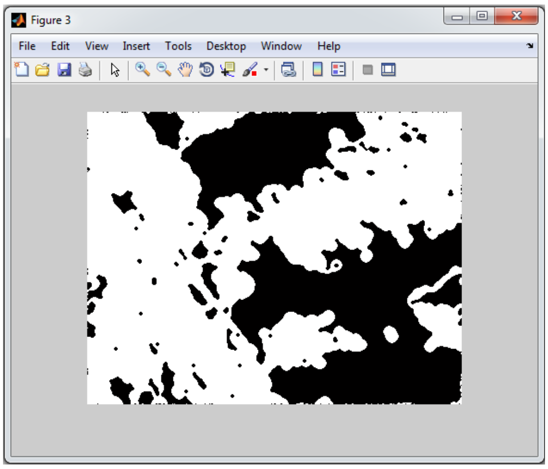


(b) Conversion to Grey Scale Image

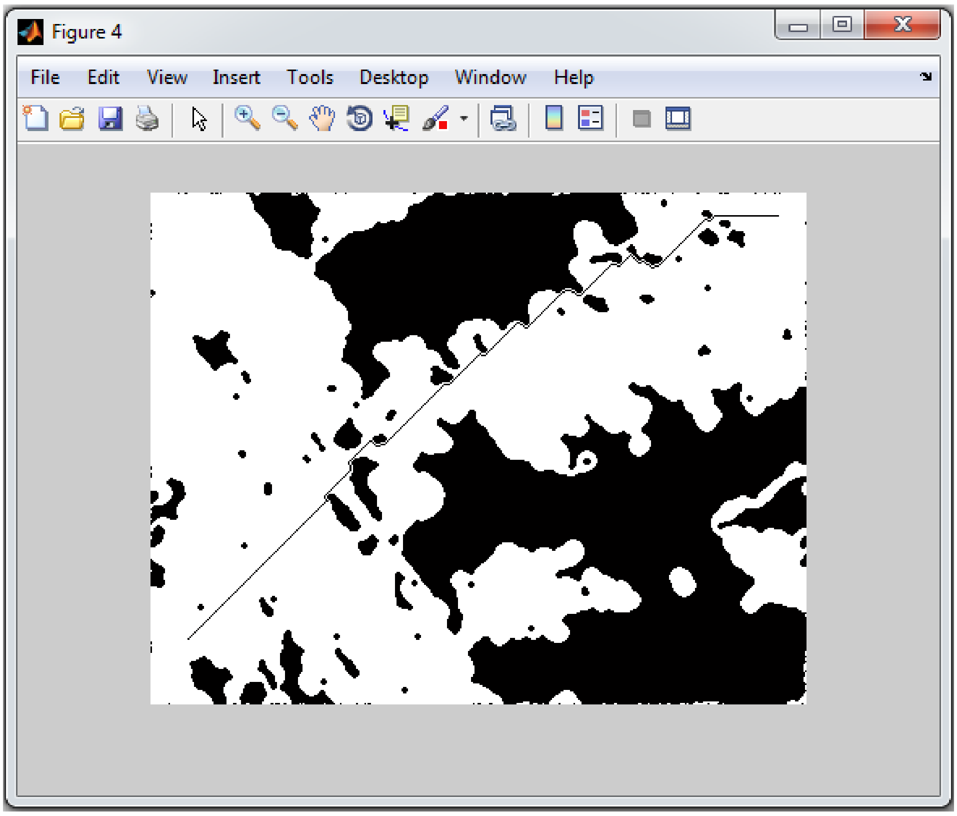
The path planning results for satellite image 5 using proposed CS-FAPP approach are illustrated in figure 4.19 (a)-(e). The size of the image is 405*316 pixels with defined start source and destination points are (24, 276) and (388, 15) respectively. On the basis of defined initial source and destination points, the proposed CS-FAPP approach has obtained the optimal path length of 376 pixels as illustrated in figure 4.19 (e).



(c) Conversion to binary image



(d) Morphologically Operated Image



(e) Obtained Optimal Path of Satellite Image 5

Figure 4.19 (a)-(e): Path Planning Results of Satellite Image 5

To analyze the performance results of proposed hybrid CS-FAPP algorithm, it is also necessary to evaluate the path planning results using another hybrid concept of CS-BAPP and individual CS, FA, and BA. These individual algorithms (CS, FA, and BA) and hybrid algorithm (CS-BAPP) also obtained the same path length as obtained using proposed CS-FAPP for the satellite images but these concepts differ in terms of simulation time and minimum number of iterations that are required to obtain the optimal path. These concepts are further elaborated in terms of performance parameters of minimum number of iterations, error rate, success rate, & simulation time. These parameters along with the evaluated results are discussed in the next section of results and discussion.

4.5. PERFORMANCE EVALUATION METRICS

The results for the optimal path planning on satellite images are evaluated in terms of evaluation parameters of minimum number of iterations, error rate, success rate, & simulation time. This experimentation has been performed on window 7 based computer system having confirmation of Intel i5 Processor and 4GB of RAM. Results are recorded using MATLAB version 8.3.0.532. The performance evaluation parameters are elaborated here:

Simulation Time*:* Simulation time can be stated as the complete amount of time taken by the algorithm to detect the collision free optimum path from defined source to destination point.

Number of iterations*:* The number of iterations can be stated as the number of repetitions of a process. All the algorithms are compared in terms of minimum number of iterations taken by each algorithm to detect the optimum path.

Success Rate*:* The term success rate can be stated as the rate of successful completion of the algorithm. Each algorithm is processed 20 times by considering the minimum number of iterations taken by individual algorithm to calculate their success rate. It can be defined as the ratio of number of successful cases to the total number of cases.

Error Rate: The term error rate can be defined as the rate of unsuccessful completion of the algorithm. It can be formulated as the ratio of number of unsuccessful/failure cases to the total number of cases. This is also evaluated based on the total 20 cases.

4.6. RESULTS AND COMPARISON

Based on the parameters discussed in the previous section, results are evaluated on MATLAB simulation tool. The results are discussed in two sections which describe results based on iteration manner and overall results.

4.6.1. Iteration Based Results

In this section, the results based on the iteration manner are evaluated. The optimal path length achieved for all the satellite images is same for the proposed CS-FAPP, hybrid algorithm of CS-BAPP, and the other individual algorithms. There is the difference of the simulation time and iteration number at which the optimal length achieved for the different algorithms. The iteration wise result with the simulation time consumed by all the algorithms for the satellite images 1-5 are illustrated in tables 4.3 - table 4.7 respectively.

The table 4.3 - table 4.7 presents the comparison of the bat algorithm (BA), firefly algorithm (FA), and cuckoo search (CS), hybrid CS-BAPP and proposed hybrid CS-FAPP algorithm in term of best path length and best simulation time with an interval of 10 iterations. The best path length and best simulation time refers to the optimized path length obtained after each 10 iterations and time taken to obtain that best path length after each 10 iteration interval respectively.

The iteration wise evaluated results for satellite image 1 presented in table 4.3 indicates the optimal path length of 320 pixels is achieved at iteration number 36 for the proposed CS-FAPP algorithm which lie in the interval of 40 iterations. The simulation time consumed for this optimal path with proposed CS-FAPP algorithm is 165-212 seconds. The other algorithms of CS, FA, BA, and CS-BAPP have achieved the best path length at the iteration interval of 70, 80, 90, and 50 respectively. These algorithms of CS, FA, BA, and CS-BAPP have achieved this optimum path length in the simulation time of 226-271 sec., 196-258 sec., 284-349 sec., and 203-266 sec. respectively.

Further, the table 4.4 indicates the results for the satellite image 2 using proposed CS-FAPP and other considered concepts. The optimal path length of 261 pixels is achieved at iteration interval of 30 for the proposed CS-FAPP algorithm with simulation time of 78-103 seconds. The other algorithms of CS, FA, BA, and hybrid CS-BAPP have achieved the best path length at the iteration interval of 40, 50, 60, and 30 respectively. These algorithms of CS, FA, BA, and CS-BAPP have achieved this optimum path length in the simulation time of 155-216 sec., 139-177 sec., 204-253 sec., and 128-181 sec. respectively.

The table 4.5 indicates the results for the satellite image 3 with optimal path length of 246 pixels. The hybrid CS-FAPP algorithm achieved the optimum path length of 246 pixels with in 27 iterations (interval 30) with the simulation time of 112-167 seconds. The other algorithms of CS, FA, BA, and CS-BAPP have achieved the best path length at the iteration interval of 50, 60, 65, and 35 respectively with simulation time of 179-230 sec., 151-195 sec., 198-264 sec., and 154-213 sec. respectively.

The table 4.6 indicates the results for the satellite image 4 with optimal path length of 318 pixels. The proposed hybrid CS-FAPP algorithm attained the optimum path length of 318 pixels at iteration interval of 40 with simulation time of 134-182 seconds. The other algorithms of CS, FA, BA, and CS-BAPP have achieved the best path length at the iteration interval of 60, 70, 70, and 40 respectively with simulation time of 196-251 sec., 173-235 sec., 257-314 sec., and 187-245 sec. respectively.

The iteration based results of satellite image 5 are illustrated in table 4.7. The proposed hybrid CS-FAPP algorithm achieved the optimum path length of 376 pixels at iteration interval of 30 with simulation time of 107-156 seconds. The other algorithms of CS, FA, BA, and CS-BAPP have achieved the best path length at the iteration interval of 50, 60, 60, and 40 respectively with simulation time of 164-228 sec., 146-191 sec., 211-269 sec., and 163-232 sec. respectively.

Table 4.3: Results of Satellite Image 1 based on Iteration Manner

No. of Iterations			10	20	30	40	50	60	70	80	90
CS	**Best Path Length**		931	860	746	633	583	467	320	NA	NA
	Best Simulation Time (in sec)	**Min.**	162	173	186	199	206	214	226	NA	NA
		Max.	190	218	231	244	251	264	271	NA	NA
FA	**Best Path Length**		884	772	691	603	559	493	409	320	NA
	Best Simulation Time (in sec)	**Min.**	127	134	152	165	171	178	188	196	NA
		Max.	151	173	189	194	211	224	239	258	NA
BA	**Best Path Length**		993	916	857	792	685	551	470	394	320
	Best Simulation Time (in sec)	**Min.**	178	190	204	228	241	254	268	273	284
		Max.	212	246	273	285	297	316	328	337	349
Hybrid CS-BAPP	**Best Path Length**		748	633	529	408	320	NA	NA	NA	NA
	Best Simulation Time (in sec)	**Min.**	143	148	162	185	203	NA	NA	NA	NA
		Max.	174	183	216	241	266	NA	NA	NA	NA
Hybrid CS-FAPP	**Best Path Length**		672	547	431	320	NA	NA	NA	NA	NA
	Best Simulation Time (in sec)	**Min.**	112	136	153	165	NA	NA	NA	NA	NA
		Max.	138	167	188	212	NA	NA	NA	NA	NA

Table 4.4: Results of Satellite Image 2 based on Iteration Manner

No. of Iterations			10	20	30	40	50	60
CS	**Best Path Length**		583	431	318	261	NA	NA
	Best Simulation Time (in sec)	**Min.**	96	110	124	155	NA	NA
		Max.	115	147	188	216	NA	NA
FA	**Best Path Length**		491	408	365	316	261	NA
	Best Simulation Time (in sec)	**Min.**	78	93	108	124	139	NA
		Max.	114	132	158	165	177	NA
BA	**Best Path Length**		624	551	418	394	316	261
	Best Simulation Time (in sec)	**Min.**	116	128	140	167	181	204
		Max.	131	148	163	198	226	253
Hybrid CS-BAPP	**Best Path Length**		478	374	261	NA	NA	NA
	Best Simulation Time (in sec)	**Min.**	69	91	128	NA	NA	NA
		Max.	93	139	181	NA	NA	NA
Hybrid CS-FAPP	**Best Path Length**		412	320	261	NA	NA	NA
	Best Simulation Time (in sec)	**Min.**	48	61	78	NA	NA	NA
		Max.	57	84	103	NA	NA	NA

Table 4.5: Results of Satellite Image 3 based on Iteration Manner

No. of Iterations			10	20	30	40	50	60	70
CS	**Best Path Length**		712	611	487	387	246	NA	NA
	Best Simulation Time (in sec)	**Min.**	96	125	142	168	179	NA	NA
		Max.	113	133	163	194	230	NA	NA
FA	**Best Path Length**		684	593	508	447	343	246	NA
	Best Simulation Time (in sec)	**Min.**	88	115	123	134	146	151	NA
		Max.	97	121	157	165	180	195	NA
BA	**Best Path Length**		803	677	504	382	291	261	246
	Best Simulation Time (in sec)	**Min.**	123	146	159	175	181	187	198
		Max.	154	173	195	216	231	253	264
Hybrid CS-BAPP	**Best Path Length**		583	454	305	NA	NA	NA	NA
	Best Simulation Time (in sec)	**Min.**	81	110	138	NA	NA	NA	NA
		Max.	92	144	182	NA	NA	NA	NA
Hybrid CS-FAPP	**Best Path Length**		518	401	246	NA	NA	NA	NA
	Best Simulation Time (in sec)	**Min.**	67	81	112	NA	NA	NA	NA
		Max.	83	106	167	NA	NA	NA	NA

Table 4.6: Results of Satellite Image 4 based on Iteration Manner

No. of Iterations			10	20	30	40	50	60	70
CS	**Best Path Length**		795	708	593	518	447	318	NA
	Best Simulation Time (in sec)	**Min.**	149	157	164	175	183	196	NA
		Max.	178	186	193	223	238	251	NA
FA	**Best Path Length**		711	687	653	594	512	455	318
	Best Simulation Time (in sec)	**Min.**	114	138	147	156	161	169	173
		Max.	142	172	189	196	215	222	235
BA	**Best Path Length**		832	761	689	604	562	485	318
	Best Simulation Time (in sec)	**Min.**	163	182	194	210	231	245	257
		Max.	197	205	234	259	273	281	314
Hybrid CS-BAPP	**Best Path Length**		621	519	443	318	NA	NA	NA
	Best Simulation Time (in sec)	**Min.**	133	154	171	187	NA	NA	NA
		Max.	164	196	227	245	NA	NA	NA
Hybrid CS-FAPP	**Best Path Length**		510	452	381	318	NA	NA	NA
	Best Simulation Time (in sec)	**Min.**	94	115	126	134	NA	NA	NA
		Max.	124	153	168	182	NA	NA	NA

Table 4.7: Results of Satellite Image 5 based on Iteration Manner

No. of Iterations			10	20	30	40	50	60
CS	**Best Path Length**		891	754	614	497	376	NA
	Best Simulation Time (in sec)	**Min.**	94	109	125	146	164	NA
		Max.	119	138	168	191	228	NA
FA	**Best Path Length**		843	752	617	561	484	376
	Best Simulation Time (in sec)	**Min.**	86	98	112	127	135	146
		Max.	101	126	138	155	172	191
BA	**Best Path Length**		926	846	712	586	423	376
	Best Simulation Time (in sec)	**Min.**	126	144	161	182	196	211
		Max.	161	192	209	223	249	269
Hybrid CS-BAPP	**Best Path Length**		642	550	451	376	NA	NA
	Best Simulation Time (in sec)	**Min.**	78	105	138	163	NA	NA
		Max.	104	143	188	232	NA	NA
Hybrid CS-FAPP	**Best Path Length**		544	483	376	NA	NA	NA
	Best Simulation Time (in sec)	**Min.**	58	81	107	NA	NA	NA
		Max.	71	124	156	NA	NA	NA

These evaluated results illustrated in table 4.3 - table 4.7 indicates that the increase of iteration number also leads to rise of the simulation time. Moreover, it can also be noticed that the size of the image and obstacles types also affect the required minimum number of iterations to obtain the optimal path length. The increase of image size also makes the path length as the source and target points in all the images are defined at the diagonal corners of image. If there will be the longer path from source to destination, then the probability of in path obstacles will also be higher. The overall it can be concluded that the increase of image size increases the occurrence of obstacles in the path which leads to increase of path length and also enhances simulation time as well.

The comparison results illustrated in table 4.3 – table 4.7 indicates the dominance of proposed hybrid CS-FAPP algorithm in terms of iterations interval. But it is also required to obtain the exact iteration number that algorithm required to obtain the optimum path length in all the satellite images. The results of minimum number of iteration required to obtain the optimum path length are evaluated in next sub-section of overall results along with the results of other evaluation metrics of success rate, error rate, and minimum & maximum simulation time.

4.6.2. Overall Results

The overall results are evaluated in terms of minimum number of iterations, success rate, error rate, and minimum & maximum simulation time. The evaluated result values for these considered parameters in terms of minimum number of iterations, simulation time, and success rate along with error rate are illustrated in table 4.8 – table 10 respectively. The evaluated results based on minimum number of iteration and simulation time are presented for the each satellite image separately. On the other hand, success rate and error rate are determined as an average for all the satellite images as these parameters defines the overall success and failure of system.

The results indicated in table 4.8 demonstrate the minimum number of iterations required by each algorithm to obtain the optimal path length of each satellite image.

In all the cases, proposed CS-FAPP approach have accomplished the optimal path length with lower most iteration number as compared to other considered concepts of CS, FA, BA, and CS-BAPP. The obtained values of minimum number of iteration with the proposed CS-FAPP algorithm for satellite images 1 – 5 are 36, 23, 27, 31, and 26 respectively. Among the other algorithms, CS-BAPP approach dominates over other individual algorithms and CS algorithm obtained the superior results over FA and BA.

Table 4.8: Overall Results based on Minimum Number of Iterations

Algorithm	**Satellite Image 1**	**Satellite Image 2**	**Satellite Image 3**	**Satellite Image 4**	**Satellite Image 5**
CS	63	35	49	52	43
FA	72	42	56	64	51
BA	87	54	61	70	58
CS-BAPP	44	29	32	38	32
CS-FAPP	36	23	27	31	26

Table 4.9: Overall Results based on Simulation Time

Algorithm		**Satellite Image 1**	**Satellite Image 2**	**Satellite Image 3**	**Satellite Image 4**	**Satellite Image 5**
CS	**Min.**	226	155	179	196	164
	Max.	271	216	230	251	228
FA	**Min.**	196	139	151	173	146
	Max.	258	177	195	235	191
BA	**Min.**	284	204	198	257	211
	Max.	349	253	264	314	269
CS-BAPP	**Min.**	203	128	154	187	163
	Max.	266	181	213	245	232
CS-FAPP	**Min.**	165	78	112	134	107
	Max.	212	103	167	182	156

The results discussed in table 4.9 extend the discussion based on simulation time. Based on the simulation time as well, proposed CS-FAPP approach outperformed over other concepts of CS, FA, BA, and CS-BAPP. The proposed hybrid CS-FAPP approach attained the results for satellite images 1-5 in simulation time of 165-212 sec., 78-103 sec., 112-167 sec., 134-182 sec., and 107-156 sec.

Table 4.10: Overall Results based on Success and Error Rate

Algorithm	Success Rate (For 20 test cases)	Error Rate (For 20 test cases)
CS	20 (100 %)	0 %
BA	18 (90 %)	10 %
FA	19 (95 %)	5 %
CS-BAPP	20 (100 %)	0 %
CS-FAPP	20 (100 %)	0 %

Further, Success rate and error rate are calculated by assuming each algorithm is processed 20 times with availability of minimum number of iterations required by each algorithm. Moreover, proposed hybrid concept achieved 100% success rate and 0% error rate with the completion of 20 successful rounds. Individual CS and hybrid CS-BAPP algorithm also achieved 100% success rate whereas individual FA and BA have error rate of 5% and 10% respectively. The failure of cases of FA and BA are due to imbalance between the exploration and exploitation of firefly agents that traps it in the local optima.

To better understand the outperformed results of hybrid CS-FAPP algorithm, these comparative results are expressed in graphical representation format. The graphical representation in terms of minimum number of iterations, minimum simulation time, maximum simulation time, success rate, and error rate are illustrated by figure 4.20-4.24 respectively.

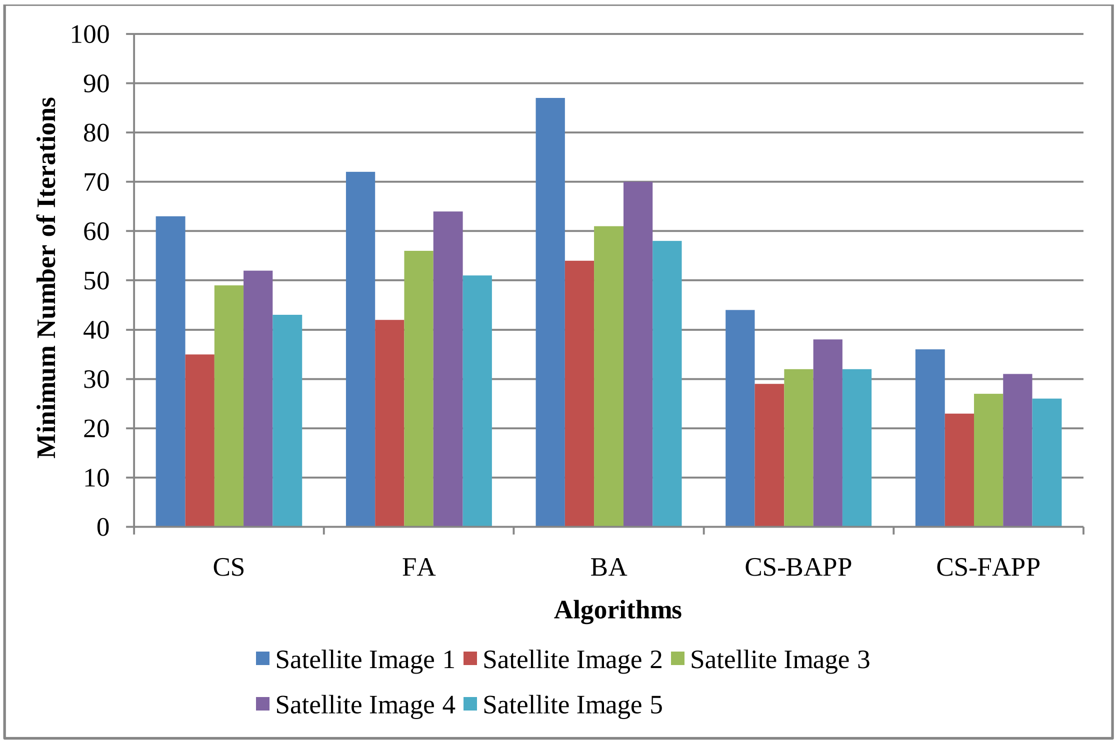


Figure 4.20: Comparison based on Minimum Number of Iterations

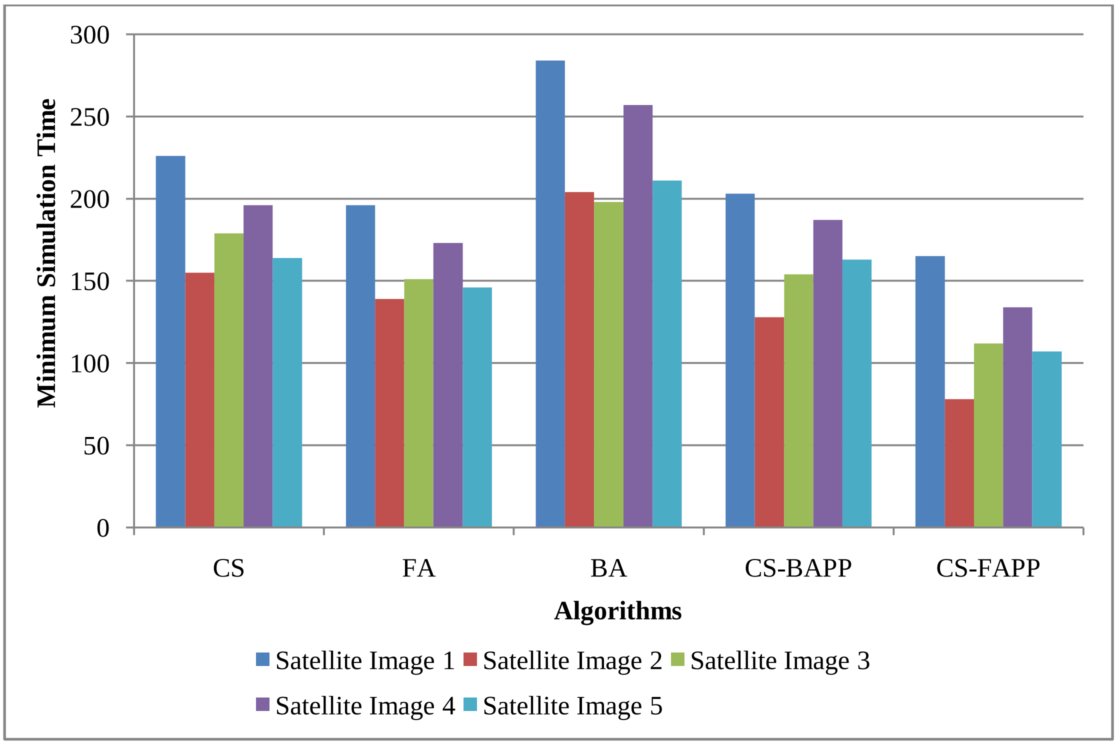


Figure 4.21: Comparison based on Minimum Simulation Time

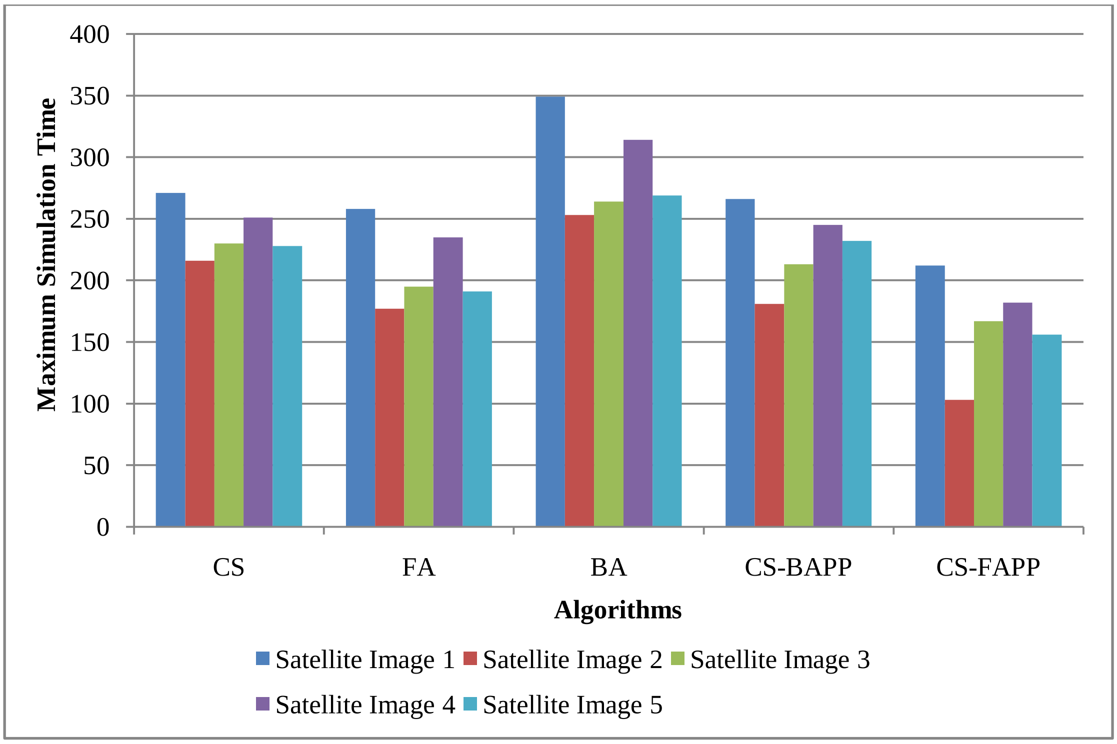


Figure 4.22: Comparison based on Maximum Simulation Time

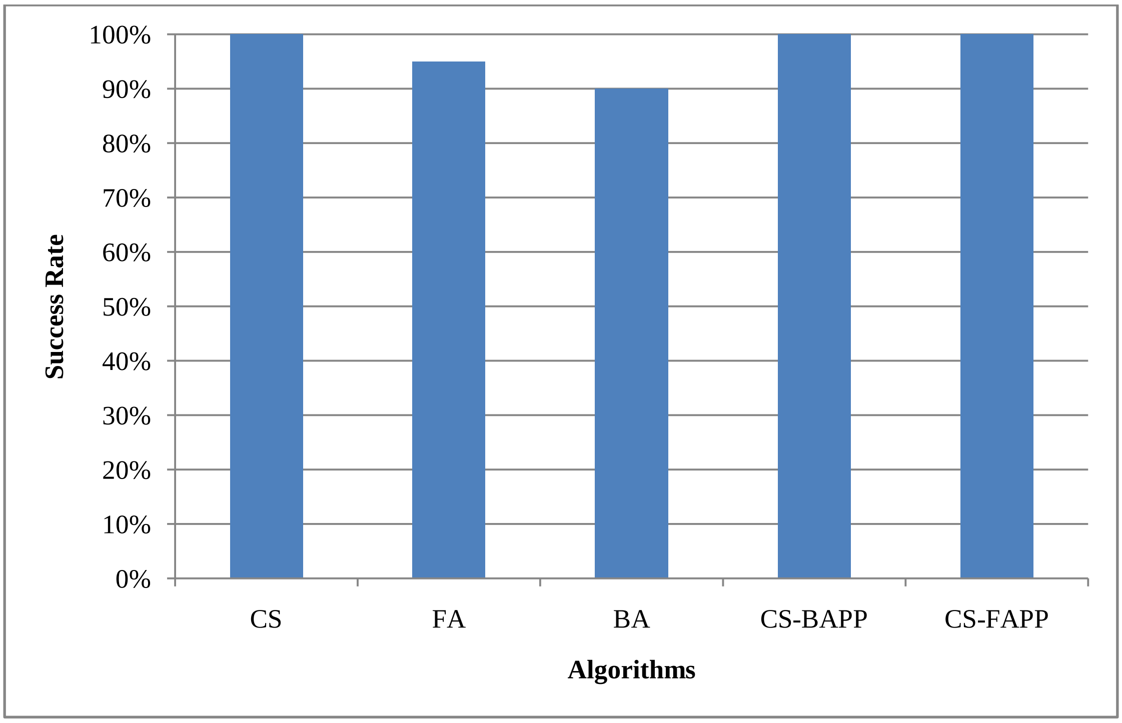


Figure 4.23: Comparison based on Success Rate

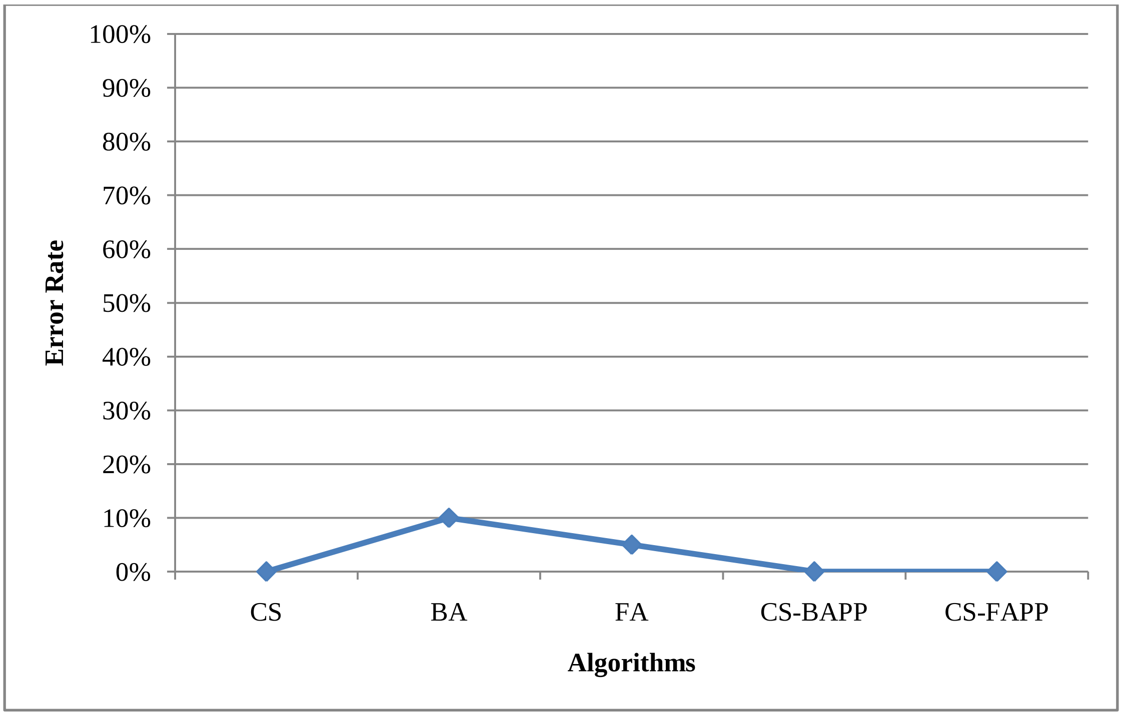


Figure 4.24: Comparison based on Error Rate

The comparison graphs illustrated in figures 4.20-4.24 indicates the superiority of proposed hybrid CS-FAPP concept in all the cases as compared to other individual and hybrid algorithms. CS-FAPP have attained 100% success rate and having minimum number of iteration with lesser simulation time in all the cases of satellite images. Other algorithms lacks than proposed hybrid CS-FAPP algorithm. However, CS-BAPP algorithm has achieved dominance results in comparison with individual concepts of CS, BA, and FA but lacks than CS-FAPP. Although, the hybrid CS-BAPP algorithm has comparable minimum and maximum simulation time to individual FA, but individual FA has 5% chances of error rate. This 5% error rate lacks the individual FA from CS-BAPP algorithm. As a whole, it can be declared that proposed hybrid CS-FAPP and dominates over other algorithms.

4.7. SUMMARY

The evaluated results using proposed hybrid CS-FAPP algorithms are discussed in this chapter. The algorithm is initially tested with benchmark functions of Rosenbrock, Michalewicz, Ackley, Easom, De Jong, Schwefel, Rastrigin, Griewank, and Shubert Function. The benchmark functions are also tested for CS-BAPP algorithms and compared with hybrid CS-FAPP and other individual algorithms. In terms of benchmark testing, CS-FAPP dominates over other algorithms. Moreover, the working example for the hybrid CS-FAPP is also presented and results are further evaluated for the path planning application with experimentation on Google based various satellite images and assessment in terms of minimum number of iterations, success rate, error rate, and minimum & maximum simulation time. Overall proposed hybrid CS-FAPP algorithm outperformed in comparison with other considered concepts.

Next chapter concludes the thesis work along with some future directions.

CHAPTER 5
CONCLUSION AND FUTURE SCOPE

The final conclusion of the research work along with future scope of the accomplished proposed work is discussed in this chapter.

5.1. CONCLUSION

In this thesis, the main focus of our research work was the optimal path planning using nature inspired concept. To accomplish this work, research objectives were designed which includes the path planning using proposed hybrid CS-FAPP (Hybrid Cuckoo Search with Firefly Algorithm for Path Planning) algorithm, evaluation of algorithm by alteration the parameters, and comparison with hybrid CS-BAPP (Hybrid Cuckoo Search with Bat Algorithm for Path Planning) algorithm. Nature inspired concepts are categorized into four categories based on geo-sciences, artificial immune system, modelization of human mind, and swarm intelligence. These concepts are discussed in chapter 1. Among these concepts swarm intelligence based CS, FA, BA are considered and defined research objectives are fulfilled. The reason for the consideration of these CS, BA, and FA algorithms is the adaptability of these concepts according to problem definition. The reason for the hybridization is to achieve more efficient results as compared to individual swarm intelligence concepts. Another reason for the hybridization is that individual BA and FA can lead to problem of trapping between local optima. This obligates to hybridize the individual concepts of BA and FA with some more efficient algorithm like CS having clever behavior and brood parasitic property to store their egg in other bird's nest. Further, hybrid concept of CS-FAPP is proposed and compared with hybrid concept of CS-BAPP algorithm for path planning application.

Path planning is key research topic in the field of robotics research, transportation, bioinformatics, virtual prototype designing, gaming, computer aided designs, and virtual reality estimation. In optimal path planning, it is important to determine the collision free optimal and shortest path. There may be various aspects to determine the optimal path based on workspace environment and obstacle types. In this research work, optimal path is determined based on the workspace environment having static obstacles and unknown environment area. There are various existing methods and algorithms for the optimal path planning with experimentation on different workspace and different obstacle handling as discussed in chapter 2.

Further, the hybrid concept of CS-FAPP is proposed and tested based on the standard benchmark functions of Rosenbrock, Michalewicz, Ackley, Easom, De Jong, Schwefel, Rastrigin, Griewank, and Shubert Function. The selection of mentioned benchmark functions is the availability of evaluated values with the CS, FA, and BA. The algorithm of CS-FAPP and CS-BAPP are tested and compared on the basis of benchmark functions along with the comparison to CS, FA, and BA. The outperformed results of these hybrid algorithms results support in favor to proceed towards the implementation of these concepts on the application of optimal path planning.

The experimentation of path planning is performed on the Google based satellite images of different regions of India. In this experimentation, the images of different sizes consists of different terrain features of water, vegetation, urbanization, hilly (rocky), and barren. The major focus area of these regions is related to vegetation based terrain features.

The hybrid CS-FAPP and CS-BAPP algorithms are applied to determine the path planning on considered satellite images. Moreover, the individual algorithms of CS, FA, and BA are also applied to determine the optimal path from defined source to destination so that the evaluated results of hybrid algorithms can be compared with the individual algorithms. The considered workspace region is assumed as collection of binary pixels values: 0 and 1 where value 1 indicates obstacle free white pixel and value 0 is black pixel with obstacle. In both the hybrid algorithms, the obstacles on the workspace are handled with cuckoo search based property with an assumption of considering obstacle as worst nest for cuckoo egg and path planning is performed by other algorithm of BA in case of CS-BAPP and FA in case of FAPP respectively. The individual concepts use their own properties to handle the present obstacles and to find the optimal path from source point to destination point in a static and unknown environment. The behavioural property of fireflies of getting attracted towards brighter firefly is considered to identify optimal path in defined workspace. The obstacles are detected based on the light intensity value. The workspace is considered as set of values varying according to the light intensity value at particular point. On the other hand, CS algorithm works with an assumption of considering obstacle as worst nest for cuckoo egg to handle the obstacles. BA uses the frequency based echolocation system to detect the obstacles and handle those obstacles.

The results for the mentioned algorithms are evaluated in terms of minimum iteration number, minimum & maximum simulation time, success rate and error rate. Initially, algorithms are tested with the iteration interval of 10 to obtain the optimal path length. The evaluated results indicate the obtained optimal path lengths of 320 pixels, 261 pixels, 246 pixels, 318 pixels, and 376 pixels for the satellite images 1-5 respectively. All the algorithms have obtained the mentioned path length but with different iterations. The minimum iterations consumed by proposed CS-FAPP approach for the satellite images 1-5 are 36, 23, 27, 31, and 26 respectively. On the other hand hybrid CS-BAPP, and individual algorithms of CS, FA, and BA algorithms also achieves the optimal path length but with different iterations and simulation time. Success rate for the CS, Hybrid CS-BAPP, and CS-FAPP algorithm is also 100% on the basis of defined 20 test rounds. On the other hand, FA and BA are determined with error rate of 5% and 10% respectively.

On the basis of mentioned results, both the CS-BAPP and CS-FAPP outperforms in comparison with individual algorithms. Moreover, the proposed CS-FAPP even performs better than another hybrid CS-BAPP algorithm. The proposed CS-FAPP approach have attained the optimized result outcomes with minimum number of iterations, lesser simulation time, and 100% success rate as compared to all the other concepts of CS-BAPP, FA, CS, and BA algorithm. Another hybrid CS-BAPP algorithm dominates over individual algorithms (CS, BA, and FA) but lacks from the proposed CS-FAPP approach.

5.2. FUTURE SCOPE

In this research work, we have introduced hybrid CS-FAPP algorithm for optimal path planning from defined source to destination. Based on the evaluated results, following directions can be considered in future:

1. The proposed CS-FAPP algorithm can be used for the path planning application with consideration of dynamic obstacle types.

2. The algorithm of FA can be replaced with some other algorithm which can attain success rate of 100% individually and can be hybridized with CS algorithm to further improve the performance of system.

REFERENCES

[1]. N. Siddique, and H. Adeli, "Nature inspired computing: an overview and some future directions", *Cognitive computation*, vol. 7, no. 6, pp. 706-714, 2015

[2]. I. J. Fister, X. S. Yang, I. Fister, J. Brest, and D. Fister, "A brief review of nature-inspired algorithms for optimization", *Elektrotehniski Vestnik*, vol. 80, no. 13, pp. 116-122, 2013.

[3]. L. Goel, D. Gupta, V. K. Panchal, and A. Abraham, "Taxonomy of nature inspired computational intelligence: a remote sensing perspective", In *2012 Fourth World Congress on Nature and Biologically Inspired Computing (NaBIC)*, pp. 200-206, IEEE, Mexico, 2012.

[4]. P. Raja, and S. Pugazhenthi, "Optimal path planning of mobile robots: A review", *International Journal of Physical Sciences*, vol. 7, no. 9, pp. 1314-1320, 2012.

[5]. J. Kennedy, "Swarm intelligence", In *Handbook of nature-inspired and innovative computing*, pp. 187-219, Springer, Boston, MA, 2006.

[6]. E. Bonabeau, D. R. D. D F. Marco, M. Dorigo, G. Théraulaz, and G. Theraulaz, *Swarm Intelligence: From Natural to Artificial Systems*, no. 1, Oxford university press, 1999.

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

[8]. X. S. Yang, and A. H. Gandomi, "Bat algorithm: a novel approach for global engineering optimization", *Engineering Computations*, vol. 29, no. 5, pp. 464-483, 2012.

[9]. M. Dorigo, and M. Birattari, "Ant colony optimization", In *Encyclopedia of machine learning*, pp. 36-39, Springer, Boston, MA, 2011.

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

[11]. X. S. Yang, and S. Deb, "Cuckoo search: recent advances and applications." *Neural Computing and Applications*, vol. 24, no. 1, pp. 169-174, 2014.

[12]. H. Shah-Hosseini, "Intelligent water drops algorithm: A new optimization method for solving the multiple knapsack problem", *International Journal of Intelligent Computing and Cybernetics*, vol. 1, no. 2, pp. 193-212, 2008.

[13]. J. Xiao, X. Zheng, X. Wang, and Y. Huang, "A modified artificial fish-swarm algorithm", In *The Sixth World Congress on Intelligent Control and Automation, 2006, WCICA 2006.*, vol. 1, pp. 3456-3460, IEEE, China, 2006.

[14]. X. S. Yang, "Firefly algorithm, Levy flights and global optimization", In *Research and development in intelligent systems XXVI*, pp. 209-218, Springer, London, 2010.

[15]. K. De Meyer, J. M. Bishop, and S. J. Nasuto, "Stochastic diffusion: Using recruitment for search", *Evolvability and interaction: evolutionary substrates of communication, signalling, and perception in the dynamics of social complexity (ed. P. McOwan, K. Dautenhahn & CL Nehaniv) Technical Report*, vol. 393, pp. 60-65, 2003

[16]. H. Ma, and D. Simon, "Blended biogeography-based optimization for constrained optimization", *Engineering Applications of Artificial Intelligence*, vol. 24, no. 3, pp. 517-525, 2011.

[17]. Y. Shi, “Particle Swarm Optimization: Developments, Applications and Resources." In *Proceedings of the 2001 Congress on evolutionary computation, 2001*, vol. 1, pp. 81-86, IEEE, South Korea, 2001.

[18]. J. Liu, and K. C. Tsui, "Toward nature-inspired computing", *Communications of the ACM*, vol. 49, no. 10, pp. 59-64, 2006.

[19]. P. Marrow, "Nature-inspired computing technology and applications", *BT Technology Journal*, vol. 18, no. 4, pp.13-23, 2000.

[20]. H. Zang, S. Zhang, and K. Hapeshi, "A review of nature-inspired algorithms", *Journal of Bionic Engineering*, vol. 7, no. 4, pp. S232-S237, 2010.

[21]. G. Rozenberg, T. Bck, and J. N. Kok, *Handbook of natural computing*, Springer Publishing Company, Incorporated, 2011.

[22]. C. Blum, and X. Li, "Swarm intelligence in optimization", In *Swarm Intelligence*, pp. 43-85. Springer, Berlin, Heidelberg, 2008.

[23]. X. S. Yang, and M. Karamanoglu, "Swarm intelligence and bio-inspired computation: an overview", In *Swarm Intelligence and Bio-Inspired Computation*, pp. 3-23, 2013.

[24]. R. S. Parpinelli, and H. S. Lopes, "New inspirations in swarm intelligence: a survey", *International Journal of Bio-Inspired Computation*, vol. 3, no. 1, pp. 1-16, 2011.

[25]. G. Beni, "From swarm intelligence to swarm robotics", In *International Workshop on Swarm Robotics*, pp. 1-9, Springer, Berlin, Heidelberg, 2004.

[26]. G. Beni, and J. Wang, "Swarm intelligence in cellular robotic systems", In *Robots and Biological Systems: Towards a New Bionics?*, pp. 703-712, Springer, Berlin, Heidelberg, 1993.

[27]. M. Dorigo, and L. M. Gambardella, "Ant colony system: a cooperative learning approach to the traveling salesman problem", *IEEE Transactions on evolutionary computation*, vol. 1, no. 1, pp. 53-66, 1997.

[28]. S. Hackwood, and G. Beni, "Self-organization of sensors for swarm intelligence." In *Proceedings 1992 IEEE International Conference on Robotics and Automation*, pp. 819-829, IEEE, France, 1992.

[29]. B. K. Panigrahi, Y. Shi, and M. H. Lim, eds, *Handbook of swarm intelligence: concepts, principles and applications*, vol. 8, Springer Science & Business Media, 2011.

[30]. J. Xie, Y. Zhou, and H. Chen, "A novel bat algorithm based on differential operator and Lévy flights trajectory", *Computational intelligence and neuroscience*, vol. 2013, pp. 1-13, 2013.

[31]. A. Ouaarab, B. Ahiod, and X. S. Yang, "Discrete cuckoo search algorithm for the travelling salesman problem", *Neural Computing and Applications*, vol. 24, no. 7-8, pp. 1659-1669, 2014.

[32]. D. Simon, M. Ergezer, D. Du, and R. Rarick, "Markov models for biogeography-based optimization." *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 41, no. 1, pp. 299-306, 2011.

[33]. M. Dorigo, and T. Stützle, "The ant colony optimization metaheuristic: Algorithms, applications, and advances", In *Handbook of metaheuristics*, pp. 250-285, Springer, Boston, MA, 2003.

[34]. A. H. Gandomi, X. S. Yang, S. Talatahari, and A. H. Alavi, "Firefly algorithm with chaos." *Communications in Nonlinear Science and Numerical Simulation*, vol. 18, no. 1, pp. 89-98, 2013

[35]. P. W. TSai, J.S. Pan, B.-Y. Liao, and S.-C. Chu, "Enhanced artificial bee colony optimization", *International Journal of Innovative Computing, Information and Control*, vol. 5, no. 12, pp. 5081-5092, 2009.

[36]. H. S. Hosseini, "Problem solving by intelligent water drops", In *IEEE congress on evolutionary computation, 2007, CEC 2007*, pp. 3226-3231, IEEE, Singapore, 2007.

[37]. I. C. Trelea, "The particle swarm optimization algorithm: convergence analysis and parameter selection", *Information processing letters*, vol. 85, no. 6, pp. 317-325, 2003.

[38]. A. Jankowski, and A. Skowron, "Wisdom technology: A rough-granular approach", In *Aspects of Natural Language Processing*, pp. 3-41. Springer, Berlin, Heidelberg, 2009.

[39]. M. Nadin, "Anticipatory computing. from a high-level theory to hybrid computing implementations", *International journal of applied research on*

information technology and computing (IJARITAC), vol. 1, no. 1, pp. 1-27, 2010.

[40]. H-J. Zimmermann, "Applications of fuzzy set theory to mathematical programming", *Information sciences*, vol. 36, no. 1-2, pp. 29-58, 1985.

[41]. J. Maiers, and Y. S. Sherif, "Applications of fuzzy set theory", *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 1, pp. 175-189, 1985.

[42]. A. Bargiela, and W. Pedrycz, "Granular computing", In *Handbook On Computational Intelligence: Volume 1: Fuzzy Logic, Systems, Artificial Neural Networks, and Learning Systems*, pp. 43-66, 2016.

[43]. Y. Yao, "A comparative study of formal concept analysis and rough set theory in data analysis", In *International Conference on Rough Sets and Current Trends in Computing*, pp. 59-68. Springer, Berlin, Heidelberg, 2004.

[44]. G. Y. Wang, Y.Y. Yao, and H. Yu, "A survey on rough set theory and applications", *Chinese Journal of Computers*, vol. 32, no. 7, pp. 1229-1246, 2009.

[45]. T. Y. Lin, Y. Y. Yao, and L. A. Zadeh, eds, *Data mining, rough sets and granular computing*, vol. 95, Physica, 2013.

[46]. J. E. Hunt, and D. E. Cooke, "Learning using an artificial immune system", *Journal of network and computer applications*, vol. 19, no. 2, pp. 189-212, 1996.

[47]. K. Miettinen, and P. P. By-Neittaanmaki, *Evolutionary algorithms in engineering and computer science: recent advances in genetic algorithms, evolution strategies, evolutionary programming, GE*. John Wiley & Sons, Inc., 1999.

[48]. F. M. Ham, and I. Kostanic, *Principles of neurocomputing for science and engineering*, McGraw-Hill Higher Education, 2000.

[49]. G. Păun, and G. Rozenberg, "A guide to membrane computing", *Theoretical Computer Science*, vol. 287, no. 1, pp. 73-100, 2002.

[50]. Y. Qin, H. Lu, Y. Xu, and H. Wang, "Saliency detection via cellular automata", In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 110-119, 2015.

[51]. Z. Ezziane, "DNA computing: applications and challenges", *Nanotechnology*, vol. 17, no. 2, pp. R27, 2005.

[52]. S. R. Desai, and R. Prasad, "A new approach to order reduction using stability equation and big bang big crunch optimization", *Systems Science & Control Engineering: An Open Access Journal*, vol. 1, no. 1, pp. 20-27, 2013.

[53]. G. PL. Walker, "Explosive volcanic eruptions—a new classification scheme", *Geologische Rundschau*, vol. 62, no. 2, pp. 431-446, 1973.

[54]. J. F. Dewey, W. C. Pitman III, W. BF Ryan, and J. Bonnin, "Plate tectonics and the evolution of the Alpine system", *Geological society of America bulletin*, vol. 84, no. 10, pp. 3137-3180, 1973.

[55]. C. G. Rossby, and R. B. Montgomery, *The layer of frictional influence in wind and ocean currents*, Massachusetts Institute of Technology and Woods Hole Oceanographic Institution, 1935.

[56]. M. L. Chanin, and A. Hauchecorne, "Lidar observation of gravity and tidal waves in the stratosphere and mesosphere", *Journal of Geophysical Research: Oceans*, vol. 86, no. C10, pp. 9715-9721, 1981.

[57]. S. M. LaValle, *Planning algorithms*, Cambridge university press, 2006.

[58]. Y. K. Hwang, and N. Ahuja, "Gross motion planning—a survey", *ACM Computing Surveys (CSUR)*, vol. 24, no. 3, pp. 219-291, 1992.

[59]. P. Bhattacharya, and M. L. Gavrilova, "Roadmap-based path planning-using the voronoi diagram for a clearance-based shortest path", *IEEE Robotics & Automation Magazine*, vol. 15, no. 2, 2008.

[60]. E. W. Weisstein, *Piano Mover's Problem*, MathWorld – A Wolfram Web Resource [online] http://mathworld.wolfram.com/PianoMoversProblem.html (accessed 15 April 2018), 1999

[61]. M. Davoodi, "Bi-objective path planning using deterministic algorithms", *Robotics and Autonomous Systems*, vol. 93, pp. 105-115, 2017.

[62]. D. Dolgov, S. Thrun, M. Montemerlo, and J. Diebel, "Path planning for autonomous vehicles in unknown semi-structured environments", *The International Journal of Robotics Research*, vol. 29, no. 5, pp. 485-501, 2010.

[63]. P. Sudhakara, and V. Ganapathy, "Trajectory Planning of a Mobile Robot using Enhanced A-Star Algorithm", *Indian Journal of Science and Technology*, vol. 9, no. 41, 2016.

[64]. A. K. Guruji, H. Agarwal, and D. K. Parsediya, "Time-Efficient A* Algorithm for Robot Path Planning." *Procedia Technology*, vol. 23, pp. 144-149, 2016.

[65]. J. BB. Silva, C. A. Siebra, and T. P. Nascimento, "A simplified cost function heuristic applied to the A*-based path planning", *International Journal of Computer Applications in Technology*, vol. 54, no. 2, pp. 96-105, 2016.

[66]. R. Geraerts, and M. H. Overmars, "The corridor map method: a general framework for real-time high-quality path planning", *Computer animation and virtual worlds*, vol. 18, no. 2, pp. 107-119, 2007.

[67]. A. Kumar, and A. Ojha, "Natural path planning using wavelet noise in static environment", *Computer Animation and Virtual Worlds*, vol. 24, no. 1, pp. 17-24, 2013.

[68]. N. Jaklin, A. Cook, and R. Geraerts, "Real-time path planning in heterogeneous environments", *Computer Animation and Virtual Worlds*, vol. 24, no. 3-4, pp. 285-295, 2013.

[69]. V. Ganapathy, P. Sudhakara, T. T. J. Jie, and S. Parasuraman, "Mobile Robot Navigation using Amended Ant Colony Optimization Algorithm", *Indian Journal of Science and Technology*, vol. 9, no. 45, pp. 1-10, 2016.

[70]. A. Hidalgo-Paniagua, M. A. Vega-Rodríguez, and J. Ferruz, "Applying the MOVNS (multi-objective variable neighborhood search) algorithm to solve the path planning problem in mobile robotics", *Expert Systems with Applications*, vol. 58, pp. 20-35, 2016.

[71]. J. Lee, and D.W. Kim, "An effective initialization method for genetic algorithm-based robot path planning using a directed acyclic graph", *Information Sciences*, vol. 332, pp. 1-18, 2016.

[72]. F. M. Santa, E. J. Gomez, and H. M. Ariza, "Global Path Planning for Mobile Robots using Image Skeletonization", *Indian Journal of Science and Technology*, vol. 10, no. 14, 2017.

[73]. M. H. Overmars, "Recent developments in motion planning", In *International Conference on Computational Science*, pp. 3-13. Springer, Berlin, Heidelberg, 2002.

[74]. P. Brož, I. Kolingerová, P. Zítka, R. A. Apu, and M. Gavrilova, "Path planning in dynamic environment using an adaptive mesh." In *Proceedings of the 23rd Spring Conference on Computer Graphics*, pp. 233-239, ACM, 2007.

[75]. O. Montiel, U. O. Rosas, and R. Sepúlveda, "Path planning for mobile robots using Bacterial Potential Field for avoiding static and dynamic obstacles", *Expert Systems with Applications*, vol. 42, no. 12, pp. 5177-5191, 2015.

[76]. W. Li, S. N. Chow, M. Egerstedt, J. Lu, and H. Zho, "Method of evolving junctions: A new approach to optimal path-planning in 2D environments with moving obstacles", *The International Journal of Robotics Research*, vol. 36, no. 4, pp. 403-413, 2017.

[77]. K. Ninomiya, M. Kapadia, A. Shoulson, F. Garcia, and N. Badler, "Planning approaches to constraint-aware navigation in dynamic environments", *Computer Animation and Virtual Worlds*, vol. 26, no. 2, pp. 119-139, 2015.

[78]. J. Zhao, Y. Guo, and X. Duan, "Dynamic Path Planning of Emergency Vehicles Based on Travel Time Prediction", *Journal of Advanced Transportation*, vol. 2017, pp. 1-14, 2017.

[79]. J. Ni, L. Wu, P. Shi, and S. X. Yang, "A Dynamic Bioinspired Neural Network Based Real-Time Path Planning Method for Autonomous Underwater Vehicles", *Computational intelligence and neuroscience*, vol. 2017, pp. 1-16, 2017.

[80]. X. Hu, L. Chen, B. Tang, D. Cao, and H. He, "Dynamic path planning for autonomous driving on various roads with avoidance of static and moving obstacles", *Mechanical Systems and Signal Processing*, vol. 100, pp.482-500, 2018.

[81]. Y-Q. Qin, D-B. Sun, N. Li, and Y-G. Cen, "Path planning for mobile robot using the particle swarm optimization with mutation operator", In *Proceedings of 2004 International Conference on Machine Learning and Cybernetics, 2004*, vol. 4, pp. 2473-2478, IEEE, Chine 2004.

[82]. P. Raja, and S. Pugazhenthi, "Optimal path planning of mobile robots: A review", *International Journal of Physical Sciences*, vol. 7, no. 9, pp. 1314-1320, 2012.

[83]. I. Duleba, and J. Z. Sasiadek, "Nonholonomic motion planning based on Newton algorithm with energy optimization", *IEEE transactions on control systems technology*, vol. 11, no. 3, pp. 355-363, 2003.

[84]. M. Moll, and L. E. Kavraki, "Path planning for minimal energy curves of constant length", In *IEEE International Conference on Robotics and Automation*, vol. 3, pp. 2826-2831, IEEE, New Orleans, 2004.

[85]. D. Kuan, J. Zamiska, and R. Brooks, "Natural decomposition of free space for path planning", In *Proceedings 1985 IEEE International Conference on Robotics and Automation.*, vol. 2, pp. 168-173, IEEE, 1985.

[86]. H. Noborio, T. Naniwa, and S. Arimoto, "A quadtree-based path-planning algorithm for a mobile robot", *Journal of Robotic Systems*, vol. 7, no. 4, pp. 555-574, 1990.

[87]. J. Barraquand, B. Langlois, and J-C. Latombe, "Numerical potential field techniques for robot path planning", *IEEE transactions on systems, man, and cybernetics*, vol. 22, no. 2, pp. 224-241, 1992.

[88]. O. Khatib, and L. M. Mampey, "Fonction decision-commande d'un robot manipulateur", *Rep.*, vol. 2, no. 7, pp. 156, 1978.

[89]. N. Hogan, "Impedance control: An approach to manipulation: Part II—Implementation", *Journal of dynamic systems, measurement, and control*, vol. 107, no. 1, pp. 8-16, 1985.

[90]. D. Koditschek, "Exact robot navigation by means of potential functions: Some topological considerations", In *Proceedings 1987 IEEE International Conference on Robotics and Automation*, vol. 4, pp. 1-6, IEEE, USA, 1987.

[91]. E. SH. Hou, and D. Zheng, "Mobile robot path planning based on hierarchical hexagonal decomposition and artificial potential fields", *Journal of Robotic Systems*, vol. 11, no. 7, pp. 605-614, 1994.

[92]. S. M. LaValle, *Rapidly-exploring Random Trees: A New Tool for Path Planning*, Technical Report (TR 98-11), Computer Science Department, Iowa State University, USA, 1998.

[93]. L.E. Kavraki, P. Svestka, J-C. Latombe, and M.H. Overmars, "Probabilistic roadmaps for path planning in high-dimensional configuration spaces", *IEEE transactions on Robotics and Automation*, vol. 12, no. 4, pp. 566–580, 1996.

[94]. L.E. Kavraki, and J-C. Latombe, "Probabilistic roadmaps for robot path planning", In *Practical Motion Planning in Robotics: Current Approaches and Future Challenges*, pp. 33–53, John Wiley, England, 1998

[95]. V. Boor, M.H. Overmars, and A.F. Van Der Stappen, "The Gaussian sampling strategy for probabilistic roadmap planners", In *Proceedings 1999*

IEEE International Conference on Robotics and Automation, IEEE, vol. 2, pp.1018-1023, 1999.

[96]. P.K. Mohanty, and D.R. Parhi, “Controlling the motion of an autonomous mobile robot using various techniques: a review”, *Journal of Advance Mechanical Engineering*, vol. 1, no. 1, pp. 24–39, 2013.

[97]. S. Sundaram, I. Remmler, and N. M. Amato, "Disassembly sequencing using a motion planning approach" In *Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation, 2001*, vol. 2, pp. 1475-1480, IEEE, South Korea, 2001.

[98]. J. Pettré, T. Siméon, and J-P. Laumond, "Planning human walk in virtual environments", In *2002. IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 3, pp. 3048-3053, IEEE, Switzerland, 2002.

[99]. N. M. Amato, and G. Song, "Using motion planning to study protein folding pathways", *Journal of Computational* Biology, vol. 9, no. 2, pp. 149-168, 2002.

[100]. O. B. Bayazit, G. Song, and N. M. Amato, "Ligand binding with OBPRM and user input", In *2001. Proceedings 2001 ICRA IEEE International Conference on Robotics and Automation*, vol. 1, pp. 954-959. IEEE, South Korea, 2001.

[101]. T. Danner, and L. E. Kavraki, "Randomized planning for short inspection paths", In *Proceedings. ICRA'00. IEEE International Conference on Robotics and Automation, 2000*, vol. 2, pp. 971-976, IEEE, USA, 2000.

[102]. M. Mareli, and B. Twala, "An adaptive cuckoo search algorithm for optimisation", *Applied Computing and Informatics,* vol. 14, no. 2, pp. 107-115, July, 2018.

[103]. G. G. Wang, S. Deb, A. H. Gandomi, Z. Zhang, and A. H. Alavi, "Chaotic cuckoo search", *Soft Computing,* vol. 20, no. 9, pp. 3349-3362, September, 2016.

[104]. M. Shehab, A. T. Khader, and M. A. Al-Betar, "A survey on applications and variants of the cuckoo search algorithm", *Applied Soft Computing*, vol. 61, pp. 1041-1059, December, 2017.

[105]. Y. Saji, and M. E. Riffi, "A novel discrete bat algorithm for solving the travelling salesman problem", *Neural Computing and Applications*, vol. 27, no. 7, pp. 1853-1866, October, 2016.

[106]. H. Wang, W. Wang, X. Zhou, H. Sun, J. Zhao, X. Yu, and Z. Cui, "Firefly algorithm with neighborhood attraction", *Information Sciences*, vol. 382, pp. 374-387, March, 2017.

[107]. S. L. Tilahun, J. M. T. Ngnotchouye, and N. N. Hamadneh, "Continuous versions of firefly algorithm: a review", *Artificial Intelligence Review*, pp. 1-48, 2017.

[108]. Google Maps, *Google Maps/Google Earth Additional Terms of Service*, January [online] https://www.google.com/intl/en-US_US/help/terms_maps.html (accessed 10 May 2018), 2018.

[109]. X. S. Yang, "Firefly algorithms for multimodal optimization", In *International symposium on stochastic algorithms*, pp. 169-178, Springer, Berlin, Heidelberg, 2009.

[110]. A. H. Gandomi, X. S. Yang, and A. H. Alavi, "Mixed variable structural optimization using firefly algorithm", *Computers & Structures*, vol. 89, no. 23-24, pp. 2325-2336, 2011.

[111]. I. Fister, I. Fister Jr, X. S. Yang, and J. Brest, "A comprehensive review of firefly algorithms", *Swarm and Evolutionary Computation*, vol. 13, pp. 34-46, 2013.

[112]. P. K. Mohanty, and D. R. Parhi, "Optimal path planning for a mobile robot using cuckoo search algorithm", *Journal of Experimental & Theoretical Artificial Intelligence*, vol. 28, no. 1-2, pp. 35-52, 2016.

[113]. Y-Q. Qin, D-B. Sun, N. Li, and Y-G. Cen, "Path planning for mobile robot using the particle swarm optimization with mutation operator", In

Proceedings of 2004 International Conference on Machine Learning and Cybernetics, 2004, vol. 4, pp. 2473-2478, IEEE, 2004.

[114]. L. Wang, Y. Liu, H. Deng, and Y. Xu, "Obstacle-avoidance path planning for soccer robots using particle swarm optimization", In *IEEE International Conference on Robotics and Biomimetics, 2006. ROBIO'06*, pp. 1233-1238, IEEE, 2006.

[115]. H. Il Kang, B. Lee, and K. Kim, "Path planning algorithm using the particle swarm optimization and the improved Dijkstra algorithm", In *PACIIA'08. Pacific-Asia Workshop on Computational Intelligence and Industrial Application*, vol. 2, pp. 1002-1004, IEEE, 2008.

[116]. Y. Zhao, and W. Zu, "Real-time obstacle avoidance method for mobile robots based on a modified particle swarm optimization", In *International Joint Conference on Computational Sciences and Optimization, 2009. CSO 2009*, vol. 2, pp. 269-272, IEEE, 2009.

[117]. N. ZakeriNejad, A. H. Bakhtiary, and MR J. Motlagh, "Use of particle swarm optimization in path planning for rough terrain", In *2010 IEEE 19th International Workshop on Robotics in Alpe-Adria-Danube Region (RAAD)*, pp. 205-209, IEEE, 2010.

[118]. H. Kundra, and M. Sood, "Cross-country path finding using hybrid approach of PSO and BBO", *International Journal of Computer Applications*, vol. 7, no. 6, pp. 15-19, 2010.

[119]. Q. Zhu, J. Hu, W. Cai, and L. Henschen, "A new robot navigation algorithm for dynamic unknown environments based on dynamic path re-computation and an improved scout ant algorithm", *Applied Soft Computing*, vol. 11, no. 8, pp. 4667-4676, 2011.

[120]. B. Englot, and F. Hover, "Multi-goal feasible path planning using ant colony optimization", In *2011 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2255-2260, IEEE, 2011.

[121]. C. Liu, Z. Gao, and W. Zhao, "A new path planning method based on firefly algorithm", In *2012 Fifth International Joint Conference on Computational Sciences and Optimization (CSO)*, pp. 775-778, IEEE, 2012.

[122]. M. Sood, and D. Kaplesh, "Cross-Country path finding using hybrid approach of PSO and BCO", *International Journal of Applied Information Systems (IJAIS)*, vol. 2, no. 1, pp. 22-24, 2012.

[123]. M. Sood, and M. Kaur, "Shortest Path Finding in Country using Hybrid approach of BBO and BCO", *International Journal of Computer Applications*, vol. 40, no. 6, pp. 9-13, 2012.

[124]. G. Wang, L. Guo, H. Duan, H. Wang, L. Liu, and M. Shao, "A hybrid metaheuristic DE/CS algorithm for UCAV three-dimension path planning", *The Scientific World Journal*, vol. 2012, pp. 1-11, 2012.

[125]. G. Wang, L. Guo, H. Duan, L. Liu, and H. Wang, "A bat algorithm with mutation for UCAV path planning", *The Scientific World Journal*, vol. 2012, pp. 1-15, 2012.

[126]. G. Wang, L. Guo, H. Duan, L. Liu, and H. Wang, "A modified firefly algorithm for UCAV path planning", *International Journal of Hybrid Information Technology*, vol. 5, no. 3, pp. 123-144, 2012.

[127]. X. Chen, Y. Kong, X. Fang, and Q. Wu, "A fast two-stage ACO algorithm for robotic path planning", *Neural Computing and Applications*, vol. 22, no. 2, pp. 313-319, 2013.

[128]. Y. Zhang, D-W. Gong, and J-H. Zhang, "Robot path planning in uncertain environment using multi-objective particle swarm optimization", *Neurocomputing*, vol. 103, pp. 172-185, 2013.

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

[130]. P. K. Mohanty, and D. R. Parhi, "Optimal path planning for a mobile robot using cuckoo search algorithm", *Journal of Experimental & Theoretical Artificial Intelligence*, vol. 28, no. 1-2, pp. 35-52, 2014.

[131]. O. Montiel, U. Orozco-Rosas, and R. Sepúlveda, "Path planning for mobile robots using Bacterial Potential Field for avoiding static and dynamic obstacles," *Expert Systems with Applications*, vol. 42, no. 12, pp. 5177-5191, 2015.

[132]. P. K. Das, H. S. Behera, and B. K. Panigrahi, "A hybridization of an improved particle swarm optimization and gravitational search algorithm for multi-robot path planning", *Swarm and Evolutionary Computation*, vol. 28, pp. 14-28, 2016.

[133]. N. Zeng, H. Zhang, Y. Chen, B. Chen, and Y. Liu, “Path planning for intelligent robot based on switching local evolutionary PSO algorithm”, *Assembly Automation*, vol. 36, no. 2, pp. 120-126., 2016.

[134]. Y. Liu, X. Zhang, X. Guan, and D. Delahaye, "Potential odor intensity grid based uav path planning algorithm with particle swarm optimization approach", *Mathematical Problems in Engineering*, vol. 2016, pp. 1-16, 2016.

[135]. A. Ayari, and S. Bouamama, "A new multiple robot path planning algorithm: dynamic distributed particle swarm optimization", *Robotics and Biomimetics*, vol. 4, no. 1, pp. 8, 2017.

[136]. K. Bibiks, Y. F. Hu, J-P. Li, P. Pillai, and A. Smith, "Improved discrete cuckoo search for the resource-constrained project scheduling problem", *Applied Soft Computing*, vol. 69, pp. 493-503, 2018.

[137]. B. K. Patle, A. Pandey, A. Jagadeesh, and D. R. Parhi, "Path planning in uncertain environment by using firefly algorithm", *Defence Technology*, vol. 14, no. 6, pp. 1-11, 2018.

[138]. U. Goel, S. Varshney, A. Jain, S. Maheshwari, and A. Shukla, "Three Dimensional Path Planning for UAVs in Dynamic Environment using Glow-

worm Swarm Optimization", *Procedia computer science*, vol. 133, pp. 230-239, 2018.

[139]. K. Althoefer, B. Krekelberg, D. Husmeier, and L. Seneviratne, "Reinforcement learning in a rule-based navigator for robotic manipulators", *Neurocomputing*, vol. 37, no. 1-4, pp. 51-70, 2001.

[140]. J. A. Sauter, R. Matthews, H. D. Parunak, and S. Brueckner, "Evolving adaptive pheromone path planning mechanisms", In *Proceedings of the first international joint conference on Autonomous agents and multiagent systems: part 1*, pp. 434-440. ACM, 2002.

[141]. M. G. Park, and M. C. Lee, "Experimental evaluation of robot path planning by artificial potential field approach with simulated annealing", In *SICE 2002. Proceedings of the 41st SICE Annual Conference*, vol. 4, pp. 2190-2195, IEEE, 2002.

[142]. M. Hamdan, and M. E. El-Hawary, "A novel genetic algorithm searching approach for dynamic constrained multicast routing", In *Canadian Conference on Electrical and Computer Engineering, 2003. IEEE CCECE 2003*, vol. 2, pp. 1127-1130, IEEE, 2003.

[143]. T-L. Lee, and C-J. Wu, "Fuzzy motion planning of mobile robots in unknown environments", *Journal of Intelligent and robotic Systems*, vol. 37, no. 2, pp. 177-191, 2003.

[144]. D. Erickson, "Non-learning artificial neural network approach to motion planning for the pioneer robot", In *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2003.(IROS 2003)*, vol. 1, pp. 112-117, IEEE, 2003.

[145]. D. Janglová, "Neural networks in mobile robot motion", *International Journal of Advanced Robotic Systems*, vol. 1, no. 1, pp. 2, 2004.

[146]. Y. Hu, and S. X. Yang, "A knowledge based genetic algorithm for path planning of a mobile robot", In *Proceedings ICRA'04. 2004 IEEE*

International Conference on Robotics and Automation, 2004, vol. 5, pp. 4350-4355, IEEE, 2004.

[147]. C-K. Loo, M. Rajeswari, E. K. Wong, and M. V. C. Rao, "Mobile robot path planning using hybrid genetic algorithm and traversability vectors method", *Intelligent Automation & Soft Computing*, vol. 10, no. 1, pp. 51-63, 2004.

[148]. X. J. Wu, Q. Li, and K. H. Heng, "Development of a vector-based fuzzy logic approach for motion planning", In *2004 IEEE Conference on Cybernetics and Intelligent Systems*, vol. 2, pp. 1380-1385, IEEE, 2004.

[149]. M. Wang and J.N.K. Liu, "Fuzzy logic based robot path planning in unknown environment", In *Proceedings of 2005 International Conference on Machine Learning and Cybernetics*, vol. 2, pp. 813-818, IEEE, 2005.

[150]. E. Masehian, and M. R. Amin-Naseri, "A tabu search-based approach for online motion planning", In *ICIT 2006. IEEE International Conference on Industrial Technology, 2006*, pp. 2756-2761, IEEE, 2006.

[151]. A. Zhu, and S. X. Yang, "A neural network approach to dynamic task assignment of multirobots", *IEEE transactions on neural networks*, vol. 17, no. 5, pp. 1278-1287, 2006.

[152]. N. B. Hui, V. Mahendar, and D. K. Pratihar, "Time-optimal, collision-free navigation of a car-like mobile robot using neuro-fuzzy approaches", *Fuzzy Sets and systems*, vol. 157, no. 16, pp. 2171-2204, 2006.

[153]. R. Araujo, "Prune-able fuzzy ART neural architecture for robot map learning and navigation in dynamic environments", *IEEE Transactions on Neural Networks*, vol. 17, no. 5, pp. 1235, 2006.

[154]. J. H. Lilly, "Evolution of a negative-rule fuzzy obstacle avoidance controller for an autonomous vehicle", *IEEE Transactions on Fuzzy Systems*, vol. 15, no. 4, pp. 718-728, 2007.

[155]. A. T. Ismail, A. Sheta, and M. Al-Weshah, "A mobile robot path planning using genetic algorithm in static environment", *Journal of Computer Science*, vol. 4, no. 4, pp. 341-344, 2008.

[156]. Z. Bi, Y. Yimin, and Y. Wei, "Hierarchical path planning approach for mobile robot navigation under the dynamic environment", In *INDIN 2008. 6th IEEE International Conference on Industrial Informatics, 2008*, pp. 372-376, IEEE, 2008.

[157]. I. Engedy, and G. Horváth, "Artificial neural network based mobile robot navigation." In *WISP 2009. IEEE International Symposium on Intelligent Signal Processing, 2009.*, pp. 241-246, IEEE, 2009.

[158]. D. R. Parhi, and M. K. Singh, "Real-time navigational control of mobile robots using an artificial neural network", In *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, vol. 223, no. 7, pp. 1713-1725, 2009.

[159]. C-H. Chao, B-Y. Hsueh, M-Y. Hsiao, S-H. Tsai, and T-H. S. Li, "Fuzzy target tracking and obstacle avoidance of mobile robots with a stereo vision system", *International Journal of Fuzzy Systems*, vol. 11, no. 3, pp. 183-191, 2009.

[160]. C-T. Cheng, K. Fallahi, H. Leung, and K. T. Chi, "An AUVs path planner using genetic algorithms with a deterministic crossover operator", In *2010 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2995-3000, IEEE, 2010.

[161]. M. Alajlan, A. Koubaa, I. Chaari, H. Bennaceur, and A. Ammar, "Global path planning for mobile robots in large-scale grid environments using genetic algorithms", In *2013 International Conference on Individual and Collective Behaviors in Robotics (ICBR)*, pp. 1-8, IEEE, 2013.

[162]. H. Duan, and L. Huang, "Imperialist competitive algorithm optimized artificial neural networks for UCAV global path planning", *Neurocomputing*, vol. 125, pp. 166-171, 2014.

[163]. I. Châari, A. Koubâa, H. Bennaceur, A. Ammar, S. Trigui, M. Tounsi, E. Shakshuki, and H. Youssef, "On the adequacy of tabu search for global robot path planning problem in grid environments", *Procedia Computer Science*, vol. 32, pp. 604-613, 2014.

[164]. L. P. Behnck, D. Doering, C. E. Pereira, and A. Rettberg, "A modified simulated annealing algorithm for SUAVs path planning", *IFAC-PapersOnLine*, vol. 48, no. 10, pp. 63-68, 2015.

[165]. Y. Lu, S. Yi, Y. Liu, and Y. Ji, "A novel path planning method for biomimetic robot based on deep learning", *Assembly Automation*, vol. 36, no. 2, pp. 186-191, 2016.

[166]. J. Lee, and D-W. Kim, "An effective initialization method for genetic algorithm-based robot path planning using a directed acyclic graph", *Information Sciences*, vol. 332, pp. 1-18, 2016.

[167]. J. Ni, L. Wu, P. Shi, and S. X. Yang, "A dynamic bioinspired neural network based real-time path planning method for autonomous underwater vehicles", *Computational intelligence and neuroscience*, vol. 2017, pp. 1-16, 2017.

[168]. D. Lin, B. Shen, Y. Liu, F. E. Alsaadi, and A. Alsaedi, "Genetic algorithm-based compliant robot path planning: an improved Bi-RRT-based initialization method", *Assembly Automation*, vol. 37, no. 3, pp. 261-270, 2017.

[169]. J. Chen, W. Jiang, P. Zhao, and J. Hu, "A path planning method of anti-jamming ability improvement for autonomous vehicle navigating in off-road environments", *Industrial Robot: An International Journal*, vol. 44, no. 4, pp. 406-415, 2017.

[170]. F. Bayat, S. Najafinia, and M. Aliyari, "Mobile robots path planning: Electrostatic potential field approach", *Expert Systems with Applications*, vol. 100, pp. 68-78, 2018.

[171]. H-Q. Min, J-H. Zhu, and X-J. Zheng, "Obstacle avoidance with multi-objective optimization by PSO in dynamic environment", In *Proceedings of 2005 International Conference on Machine Learning and Cybernetics, 2005*, vol. 5, pp. 2950-2956, IEEE, 2005.

[172]. M. M. Mohamad, M. W. Dunnigan, and N. K. Taylor, "Ant colony robot motion planning", In *The International Conference on Computer as a Tool, 2005, EUROCON 2005*, vol. 1, pp. 213-216, IEEE, 2005.

[173]. M. M. Mohamad, N. K. Taylor, and M. W. Dunnigan, "Articulated robot motion planning using ant colony optimisation", In *2006 3rd International IEEE Conference on Intelligent Systems*, pp. 690-695, IEEE, 2006.

[174]. H. Mei, Y. Tian, and L. Zu, "A hybrid ant colony optimization algorithm for path planning of robot in dynamic environment", *International Journal of Information Technology*, vol. 12, no. 3, pp. 78-88, 2006.

[175]. M. A. P. Garcia, O. Montiel, O. Castillo, and R. Sepúlveda, "Optimal path planning for autonomous mobile robot navigation using ant colony optimization and a fuzzy cost function evaluation", In *Analysis and Design of Intelligent Systems using Soft Computing Techniques*, pp. 790-798, Springer, Berlin, Heidelberg, 2007.

[176]. M. A. P. Garcia, O. Montiel, O. Castillo, R. Sepúlveda, and P. Melin, "Path planning for autonomous mobile robot navigation with ant colony optimization and fuzzy cost function evaluation", *Applied Soft Computing*, vol. 9, no. 3, pp. 1102-1110, 2009.

[177]. H. C. Huang, and C-C. Tsai, "Global path planning for autonomous robot navigation using hybrid metaheuristic GA-PSO algorithm", In *2011 Proceedings of SICE Annual Conference (SICE)*, pp. 1338-1343, IEEE, 2011.

[178]. M-Y. Ju, S-E. Wang, and J-H. Guo, "Path planning using a hybrid evolutionary algorithm based on tree structure encoding", *The Scientific World Journal*, vol. 2014, pp. 1-8, 2014.

[179]. M. A. Contreras-Cruz, V. Ayala-Ramirez, and U. H. Hernandez-Belmonte, "Mobile robot path planning using artificial bee colony and evolutionary programming", *Applied Soft Computing*, vol. 30, pp. 319-328, 2015.

[180]. M. R. Panda, R. Priyadarshini, and S. K. Pradhan, "Autonomous mobile robot path planning using hybridization of particle swarm optimization and Tabu search", In *2016 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC)*, pp. 1-7, IEEE, 2016.

[181]. P. K. Das, H. S. Behera, S. Das, H. K. Tripathy, B. K. Panigrahi, and S. K. Pradhan, "A hybrid improved PSO-DV algorithm for multi-robot path planning in a clutter environment", *Neurocomputing*, vol. 207, pp. 735-753, 2016.

[182]. J-H. Zhang, Y. Zhang, and Y. Zhou, "Path Planning of Mobile Robot Based on Hybrid Multi-Objective Bare Bones Particle Swarm Optimization With Differential Evolution", *IEEE Access*, vol. 6, pp. 44542-44555, 2018.

[183]. X-S. Yang, "Firefly algorithm, Levy flights and global optimization", In *Research and development in intelligent systems XXVI*, pp. 209-218, Springer, London, 2010.

[184]. X-S. Yang, "A new metaheuristic bat-inspired algorithm", In *Nature inspired cooperative strategies for optimization (NICSO 2010)*, pp. 65-74, Springer, Berlin, Heidelberg, 2010.

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

[186]. D. H. Ackley, "An empirical study of bit vector function optimization", *Genetic algorithms and simulated annealing*, pp. 170-204, 1987.

APPENDIX A: LIST OF PUBLICATIONS

Monica Sood, Ashish Kr Luhach, and Vinod Kumar Panchal, “Analysis of Nature Inspired Intelligence in the domain of Path Planning and searching in Cross Country with consideration of various constrained parameters”, *International Journal of Engineering and Technology ISSN: 0975-4024*, Vol. 8, No.1, pp. 92-97, 2016 **(Scopus Indexed)**.

Monica Sood, and Vinod Kumar Panchal, “Meta-heuristic techniques for path planning: recent trends and advancements”, *International Journal of Intelligent Systems Technologies and Applications, ISSN: 1740-8873*, Online Available in Upcoming Article List, 2018, **(Inderscience, Scopus Indexed)**.

Monica Sood and Vinod Kumar Panchal, “Optimal path planning with hybrid firefly algorithm and cuckoo search optimisation”, *International Journal of Advanced Intelligence Paradigms, ISSN: 1755-0394*, Online Available in Upcoming Article List, 2018, **(Inderscience, Scopus Indexed)**.

Monica Sood, Sahil Verma, Vinod Kumar Panchal, and Kavita, “Optimal Path Planning using Hybrid Bat Algorithm and Cuckoo Search”, *International Journal of Engineering & Technology (UAE) ISSN: 2227-524X and Fenyman100 4th International Conference on Computing Sciences*, 2018, **(Scopus Indexed)**

Monica Sood, Sahil Verma, Vinod Kumar Panchal, and Kavita, “Optimal Path Planning using Swarm Intelligence based Hybrid Techniques”, *Journal of Computational and Theoretical Nanoscience, ISSN: 1546-1955, JCTN-SICC-19-0003(Special Issue)*, **(Scopus Indexed)** (Accepted).

Monica Sood, Sahil Verma, Vinod Kumar Panchal, and Kavita, “Analysis of Computational Intelligence Techniques for Path Planning”, *International Conference on Computational Vision and Bio Inspired Computing (ICCVBIC 2018)*, **(Scopus Indexed Springer Conference)** (Accepted and Presented).