Multi-robot cooperation method for localizing odor sources based on a niching Firefly algorithm

A Dissertation

submitted

By

Noktienla Aier

(Regd. No. 11307319)

To

Department of Computer Science and Engineering

In partial fulfilment of the Requirement for the

Award of the Degree of

Master of Technology in

Information technology

Under the guidance of

Mrs. Monica Sood

(Assistant Professor, Lovely Professional University)

(May 2015)

PAC APPROVAL

INDIA'S LARGEST UNIVERSITY*
LOVELY PROFESSIONAL UNIVERSITY
Transforming Education, Transforming India

School of: Computer Science and Technology

DISSERTATION TOPIC APPROVAL PERFORMA

Name of the Student: NOKTIENLA . AIER Registration No: 11307319

Batch: 2013 Roll No. A32

Session: 2014-2015 Parent Section: K2308

Details of Supervisor: Designation: AP

Name MONICA Qualification: M.TECH

U.ID 14858 Research Experience: 4 years

SPECIALIZATION AREA: Computational Intelligence **(pick from list of provided specialization areas by DAA)**

PROPOSED TOPICS

1. Swarm Intelligent Techniques
2. Swarm Intelligent Algorithms include Firefly algorithm, Bat Algorithm, Altruism Algorithm etc
3. Working on the Image processing techniques

Monica Sood
Signature of Supervisor

PAC Remarks:

First topic approved.

11,29

APPROVAL OF PAC CHAIRPERSON: **Signature:** 30/9/14 **Date:**

*Supervisor should finally encircle one topic out of three proposed topics and put up for approval before Project Approval Committee (PAC)
*Original copy of this format after PAC approval will be retained by the student and must be attached in the Project/Dissertation final report.
*One copy to be submitted to Supervisor.

ABSTRACT

Problem for odor sources localization, a multi-robot cooperation method based on a niching Firefly algorithm. The proposed algorithm is Firefly algorithm. In this method, a robot will act as a particle; particles randomly distributed in the niche will start to search the odor sources. During the movement phase, attractiveness and brightness will determine where a firefly with less brightness will be attracted to the brightest one. As brightness is directly associated to the odor source, multi-robot will localized the odor sources based on this scenarios, also collision rate will be calculated during the searching processes. Results will be based on the number of iterations when all of the odor sources are localized and Average Time consumption and fitness value are calculated in comparison with Particle Swarm optimization, for an efficient result.

CERTIFICATE

This is to certify that **Noktienla Aier** has completed M. Tech dissertation titled **Multi-robot co-operation method for localizing odor sources based on a niching Firefly Algorithm** under my guidance and supervision. To the best of my knowledge, the present work is the result of her original investigation and study. No part of the dissertation has ever been submitted for any other degree or diploma. The dissertation is fit for the submission and the partial fulfilment of the conditions for the award of M. Tech Computer Science and Engineering.

Date:

Signature of Guide

Name: Mrs. Monica Sood

UID: 14858

ACKNOWLEDGEMENT

It is with deep gratitude and appreciation that I acknowledge the professional guidance, support, advice and understanding of Assistant Professor, Mrs. **Monica Sood** of Department of Computer Science and Engineering, Lovely Professional University, Jalandhar, Punjab throughout the dissertation session. Her knowledge in “Swarm Intelligence” helped me in finding appropriate details to my study in this field. I would like to thank the **Project Approval Committee members** for their valuable comments and discussions. I would also like to extend my gratitude to **Lovely Professional University** for their aid in academic studies and for allowing me to take parlance in this study. Lastly, I would also like to thank God for his blessing and guidance.

(Noktienla Aier)

DECLARATION

I hereby declare that the dissertation entitled **Multi-robot cooperation method for localizing odor sources based on a niching Firefly algorithm** for the M.Tech degree is entirely my original work and all the ideas and references have been dully acknowledged. It does not contain any work for the award of any other degree or diploma.

Date: ___________

Investigator: Noktienla Aier

Registration No: 11307319

TABLE OF CONTENTS

CONTENTS PAGE NO.

1. INTRODUCTION..1
1.1 Swarm Intelligence...1
1.2 Swarm Intelligence in Optimization...3
1.3 Swarm Intelligence Algorithms..4
1.4 Firefly Algorithm..6
1.4.1 Advantages of Firefly Algorithm...7
1.5 Swarm Intelligence Advantages...7
1.6 Swarm Intelligence Limitations…………...7
1.7 Niche Techniques……………………………………………………………..8
1.8 Autonomous Mobile Robots…………………………………………………8
1.8.1 Application Areas……...9
1.8.2 Robot Navigation..9
2. REVIEW OF LITERATURE...11
3. PRESENT WORK………..15
3.1 Problem Formulation………………………………………………………....15
3.2 Objectives………………………………………………………………………15
3.3 Methodology……………………………………………………………………16
3.4 Matlab……………………………………………………………………………...20
4. RESULTS AND DISCUSSIONS…………...22
5. CONCLUSION AND FUTURE SCOPE……...42
6. LIST OF REFERENCES..43
7. APPENDIX………………………………………………………………………46

LIST OF FIGURES

1.1 Swarm theory and Stigmergy…………………………………………………4
3.1 Niching Firefly Algorithm..19
4.1 A Simple GUI……………………………………………………………..22
4.2 3-D dimensional figure…. ………………………………………………..23
4.3 Odor Localization……………………….……………………………...24
4.4 Intensity of Flies…………………………….……………………………25
4.5 Fitness value……………………….………………………………………...25
4.6 Time Consumption……………….………………………………………26
4.7 Collision rate……………………………………………………………...26
4.8 Odor Localization…………………………………………………………27
4.9 Intensity of Flies…………………………………………………………..27
4.10 Fitness value………………………………………………………………28
4.11 Time Consumption………………………………………………………..29
4.12 Collision rate……………………………………………………………...29
4.13 Odor Localization for case-1……………………………………………...30
4.14 Fitness value for case-1…..…………………………………………...…..31
4.15 Intensity of flies for case-1………………………..………………………32
4.16 Time Consumption for case-1…………………………………………….32
4.17 Collision rate for case-1…………………………………………………...33
4.18 Odor Localization for case-2……………………………………….......33
4.19 Fitness value for case-2….………………………………………………..34
4.20 Intensity of flies case-2…………………………………………...……35
4.21 Time Consumption for case-2…………………………………………….35
4.22 Collision rate for case-2…………………………………………………..36
4.23 Odor Localization for case-3……………………………………………..36
4.24 Fitness value for case-3….………………………………………………….37
4.25 Intensity of flies for case-3……………………………………….……38
4.26 Time Consumption for case-3…………………………………………....38
4.27 Collision rate for case-3……………………...…………………………...…39
4.28 Graphical presentation of our approach and PSO……………………………..41

LIST OF TABLES

1. Firefly Algorithm……………………………………………………………….40
2. Particle swarm optimization……………………………………………………….40

CHAPTER 1
INTRODUCTION

With various changes and new things being done results in some new invention or improvement to an existing one are parts of daily life. Thus, all these constitute the research work. Research is done to reaffirm the results of previous work, solving new or existing problems which are beneficial not only to the computer professionals but also to the whole world. Research defines understanding or gaining knowledge about a topic or issue with the available data or information. Hence, this research falls under an area called Swarm Intelligence.

1.1 Swarm Intelligence

Swarm Intelligence is defined as:-"A self-organized systems, and emergent collective intelligence groups of simple agents."

Swarm Intelligence being inspired from the nature and biological systems are a group of population where simple agents, work independently, interacting locally with one another. There is no concentration on how the agents will behave, but just simple rules with certain degree of random and thus interacting with one another leads to the emergence of 'Intelligent' global behaviour. It gets its principal from various living organisms. Based on the behaviour of the swarms, multiple algorithms are prepared to solve real world problems. It was coined by Gerardo Beni and Jing Wang in 1989. The term Swarm Intelligence defines methods or algorithms by observing the behaviour of the real organisms. Swarm Intelligence possesses certain additional properties include: robustness, co-ordination, flexibility, efficiency, parallelism and reliability.

Swarm Intelligence governs certain principles:

- Emphasis positive feedback
- Removes poor solutions i.e., negative feedback
- Randomness, regardless of whether the solutions is good or poor quality
- Shared Interactions with a key to build the best solutions.

Swarm Intelligence aims at following:

- Indirect Interactions
- Distribution of work load and task management
- Decentralized approach, parallelism and no dependency

- Faster and saves time
- Provides Optimization and scalability
- Distribution system of interacting autonomous systems
- Co-operative work formation

Therefore SI can be applied to multiple applications:

- Data Mining
- Job Scheduling
- Telecommunications
- Transportation
- Entertainment
- Biometrics
- Military applications

Swarm Intelligence are a collection of intelligent systems includes: dispersed computation, contact between the swarm can be direct or indirect. The swarm agents are fortified with simple computational abilities. Adapts robustness and adaptiveness properties. Dynamic methods are structured, multistability coexistence are a part of global optimization. Swarm intelligence agents move randomly in the search space, searches and finds food source. The agents explore and exploit the environment searches for food until a more profitable food source has been depleted. This means that in a dynamic circumstances agents allow to load the information about food exploited and thereby exploration of new sites at that same time. Changing conditions is a key trade-off between exploitation and exploration. Exploitation means making use of current information and exploitation means searching for new information. The swarm agents such as ant colony, agents communicate with each other to obtain the selection of the best nest. Based on two conditions: firstly, either the ant colony moves to the new nest as the old one is destroyed or may be previous colony has developed so much that, section of it with one or more queen is sent off to start a new colony or nest. They move in groups searching for food source. It is a self-organised group movement such as swarm of insects or schools of fish. For the group movement phase, swarm intelligence techniques, Glowworm inspired based on fireflies, produce a luminescence quantity of light. Therefore, the glow worm with more brightness will get attracted by all other glow worm which will have less quantity of brightness. As such all the glow worm moves together in a group in order to obtain the best optimal solution. In glow worm the brighter one will act as a leader and all other will follow its path.

1.2 SWARM INTELLIGENCE IN OPTIMIZATION

Optimization techniques have become progressively common for decades. They imitate the behaviour of schools of fish, flocks of birds and social insects. Advantages of optimization techniques over traditional techniques are their robustness and flexibility. These properties make swarm intelligence more powerful to tackle all complex problems. Optimization techniques can solve to evaluate global best optimum solutions. It is also used to solve real world problems. In early 1990's ant colony optimization was acquainted as a technique for combinatorial optimization. This technique was inspired by real ant colonies. Whereas Particle Swarm Optimization was acquaint for continuous optimization, inspired by bird flocking. Industrial world proposed optimization techniques for complex problems. It can also solve problems related to computational biology including protein folding, medical fields, in biochemistry, also travelling Salesman problem can be solved by optimization techniques. Several algorithms are deployed for optimization technique with their own way of finding optimum solutions. Stability, quality, adaptability is quite the principles of Swarm intelligence. Stability characterizes the cohesiveness of the swarm as it moves. Distance and relative velocity defines stability of swarm. Attractant and repellant properties are embedded so the swarm will move in a group and avoid repellants.

In the fields of aircraft design, automobile design, finance multiobjective optimization problems can be found. Swarm technique such as ant colony optimization and particle swarm optimization are used to solve multi-objective problems. Particle Swarm optimization perform well for static problems whereas ant colony optimization perform well for sequencing, scheduling, discrete and graph coloring problems. Also evolutionary algorithms are used. This algorithm solves travelling salesman and vehicle routing problem. Swarm intelligence can be used to obtain routing in mobile ad-hoc networks. Routing means sent data packets from a source node to the destination node. Due to the agility of the network elements and lack of key control, routing algorithms are adapted to work in a self-organizing way. Ant colony optimizations are proposed to acquire these characteristics.

Stigmetry is a process of secondary collaboration between agents. It is a figure of self-organization. In stigmetry one agents alter environments and the other agents reply to the alter. Even without straight intercommunication with the agents it produces detailed understanding formation. Supports effective intercommunication between rigid single agents.

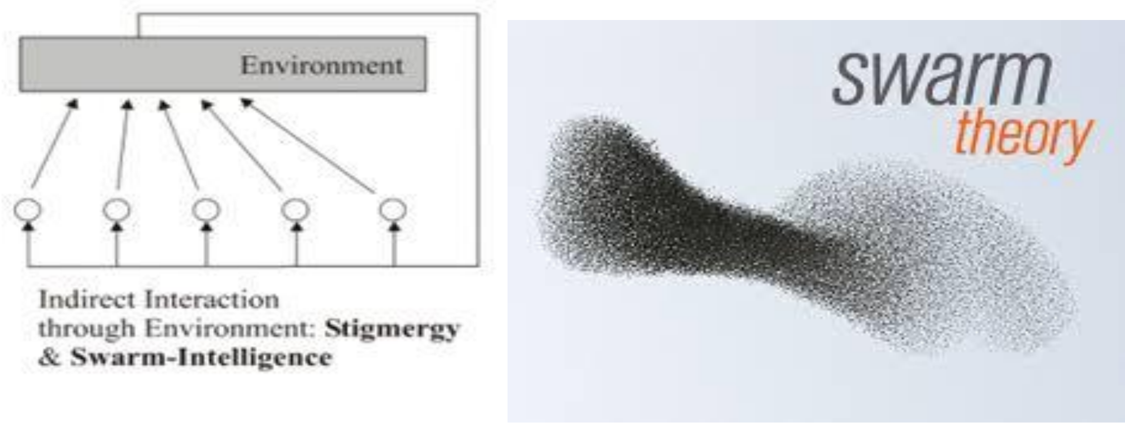

Fig: 1.1 Stigmergy and swarm theory

1.3 SWARM INTELLIGENCE ALGORITHMS

(i) Particle Swarm Optimization

Particle Swarm Optimization is an algorithm, where the particles are randomly distributed in the search space, determines the particle velocity and position. In the movement phase, every particle is known by its local best position, where the particles are in random search in order to obtain its best known position. Update the best position and thus all the swarms move towards the best solutions.

(ii)Artificial Bee Colony

Artificial Bee Colony algorithm is based on the behaviour of the bees. Three phases of the algorithm are scout bee, onlooker and the employed bee. The employed bee finds a food source and dances at this, while the onlooker bee based on the dance moves of the employed bees, chooses the food source. The abandoned food sources thus become a scout. The number of food corresponds to the number of employed bees in the colony. And the abandoned food source is replaced by the new food resource, hence the optimal food source is found.

(iii) Bat Algorithm

This algorithm is based on the echolocation behaviour of micro bats. The bat flies randomly with a certain velocity at position with varying frequency and loudness. Update frequency, loudness and pulse emission rate when the bat searches and finds its prey. Search continues until a certain stop criterion is obtained and the best solution is found.

(iv) Cuckoo Search

Developed by Xin-She Yang and Suash Deb in December 2009, this algorithm is based on the breeding parasitism of the cuckoo bird. Every single cuckoo bird will lay eggs in the nests of other host birds. The host birds will either abandon its nests or accept it. Some cuckoo can even impersonate colors of the host bird. The eggs with high value are passed to the next generation or at the highest generation, the best egg will survive and all the rest of the eggs will be abandoned. Its simplicity is an advantage to this algorithm.

(v) Ant Colony Optimization

Developed by Marco Dorigo, this algorithm is based on the behaviour of ants in search for food. When a food source is located, it persuades other ants in the direction of the food source. This interaction is via a pheromone path. The ants on locating a food source, moves food item to their nest and places pheromone along the path. This pheromone concentration will attract all the ants to follow the path resulting in the shortest path i.e., the optimal path. Higher the pheromone concentration, higher the possibility of being selected. Applicable in telecommunication networks, scheduling and graph coloring. Advantages includes inherent parallelism, use in dynamic applications. Disadvantages include difficulty in theoretical analysis rather than experimental, uncertainty in time convergence and probability distribution changes by iteration.

(vi) Glowworm Swarm Optimization

Glowworm belongs to a family of beetles known as fireflies and produces natural light (bioluminescence) that is used as a signal to attract a mate. The light is also used to attract prey. Based on the algorithm, all the glowworm contains an equal quantity of luciferin. Each iteration consists of a luciferin-update phase followed by a movement-phase. Initially all the glowworm start with the same luciferin value, value changes according to the function values at their current positions, resulting in addition to its previous luciferin level. During the movement phase each glowworm decides, using a probalistic mechanism, to move towards a neighbour that has a luciferin value more than its own. That is, they are attracted to neighbours that glow brighter. Glowworm swarm optimization is applicable for pattern recognition, document categorization and bioinformatics applications.

(vii) Intelligent Water Drops

Another algorithm which is swarm-based optimization. Water drops determine the rivers route to the destination. This algorithm determines lowest soil as the optimal path. Different applications for IWD-based algorithm include vehicle routing problem, feature selection, scheduling, and economic load dispatch and data aggregation in wireless sensor networks.

(viii) Krill Herd Algorithm

It is a nature-inspired metaheuristics algorithm. It copies the herding behaviour of krill individuals. The minimum distance from food and highest density of the herd determines its objective function value.

(ix) Fish School Search

Fish School Search is another algorithm based on the common activities of schools of fish. It is appropriate for optimization in a search space. It is based on feeding, swimming and breeding. Each Fish enact local search and this social information is accumulated by the school. This algorithm is studied on all phases with various methods and standards to solve this optimization problem. Local calculations, scalability, minimum contact between the neighbours, multiple ways of storing the information, self-controlling functionality are the principles of fish school search.

From several algorithms of Swarm Intelligence, firefly algorithm is chosen as the algorithm used for this research work. It is a nature inspired meta-heuristics algorithm. Based on the flashing behaviour of fireflies. Firefly algorithm can outperform multiple meat heuristics algorithm with its intermittent searching tactic. Therefore the research work is based on firefly algorithm for localizing odor sources using multi robot cooperation method in a niching environment.

1.4 FIREFLY ALGORITHM

Firefly Algorithm is a meta-heuristics algorithm influence from the flashing actions of fireflies. It is a swarm intelligence based algorithm which acts as a multiple interacting agents. The agents in this algorithm are randomly distributed into the workspace. The agents will carry a luminescence quantity for mating of fireflies, also explores and searches for prey randomly. Impersonating signal system to captivate other fireflies is the main objective of this algorithm. Fireflies are unisex so they get attracted to one another. A brighter firefly will attract the other firefly. Brightness depends on the distance between the fireflies; increase in the distance will decrease its brightness and attractiveness. Attractiveness is directly proportional to the brightness of the firefly. The brightness is determined by the landscape of the objective function. Fireflies move randomly if there is no particular brighter firefly.

1.4.1 ADVANTAGES OF FIREFLY ALGORITHM

(i) As firefly algorithm is based on the attractiveness, decreases with increase in the distance between the two fireflies, this leads to the sub-division of the population into sub-groups each swarm around the search space to find the local optimal position. Thus, finding the best possible position among all these position.

(ii)With the advantages of sub-divisional process allows firefly to find its optimal position simultaneously in the environment even though the population size is higher than the number of modes. Thus, its intermittent searching tactic out-performs other meta-heuristics swarm intelligence algorithm.

1.5 SWARM INTELLIGENCE ADVANTAGES

- **Scalability:** Swarm Intelligence systems are highly scalable. Because of their grouping agents, effective performances are maintained.
- **Adaptability:** Because of their inherit auto-configuration and self-organization capabilities, swarm intelligence systems adapt well to environment changes.
- **Collective Robustness:** Swarm Intelligence are more robust as they work mutually and not as an individual. Credentials of fault tolerance is remarkably high, therefore system with a single point of failure will jeopardize the entire system into a complete failure.
- **Individual Simplicity:** Swarm Intelligent systems have individual simplicity but they are limited. These limited behavioural rules are competent to collectively merge the individual to a group of systems.

1.6 SWARM INTELLIGENCE LIMITATIONS

- **Time-Critical Applications:** Time-Critical applications are invalid for swarm intelligence because their pathways are neither predefined but rather developing. Time-Critical applications require on-line control of systems and time critical decisions.
- **Parameter Tuning:** Swarm Intelligence parameters are problem dependent and are adaptively re-programmed on run time. Therefore, parameter tuning is taken as one of the drawbacks of Swarm intelligence.
- **Stagnation:** Swarm Intelligence is self-organized systems but coordination of the swarms is not centralized. Because of such reasons, stagnation problem suffer in swarm intelligence systems.

1.7 NICHE TECHNIQUES

A niche technique is capable of solving maxima optimization problems. Niche technique can conserve the variety of population. Niche techniques include fitness sharing methods where every individual will share a fitness function to define the similarity phase in a population. In clustering methods, individuals merge into niches using clustering analysis. In crowding methods, similarity is reserve to maintain the variety of a population. Objective fitness stretching method is to find the best optimal solution in the search space. Speciation method form species depending the fitness value of an individual and the radius of a species. Multi-swarm methods deploy a swarm in order to search for optimal particles. Niching techniques is used in solving multi-modal optimization.

1.8 Autonomous Mobile Robots

A recent advance in robots for localizing odor sources has been tremendously increasing using principles of swarm intelligence. In method to localize certain measurements a robot has access to feedback around its impelling actions and the condition of the surrounding around it. Therefore the robot has to terminate its location precisely. In addition, robots are becoming more and more mobile and autonomous. They are mobile i.e. the mobility of the robots is the level to which the robots can move freely around the environment. Autonomy of robots depends on the limit a robot will have preeminent knowledge from the surroundings to attain its tasks. The robots will further have three group of autonomy: non-autonomous, semi-autonomous and fully-autonomous robots.

- Non-autonomous robots are totally being controlled by the humans. The robot intelligence will demonstrate the commands received from the individual.
- Semi-autonomous robots can be in both cases, either control by the individual or steer by themselves. There are situations where the human or individual cannot take full control over the robot, thus navigating them by adjusting to the area.
- Fully-autonomous robots are fully steered by themselves. Fully-autonomous robots are suitable for intelligent act and motion, keeping aside individual relation to achieve its tasks.

Besides whether its autonomy or not autonomy depends on the desired situation. The only matter is we want a robot to achieve some general goal specified.

1.8.1 Application Areas

The usage of robots not only in industrial perspective but are getting more popularity in wider perspectives also. Autonomous robots are robots that are mostly used in environments where individual and humans find it difficult to research or probe. Researches on hazardous sites like radioactive atmosphere, space mission are quite an example.

Autonomous robots can be used in areas like undersea, land and in air. In undersea robot vehicle acts as an autonomous to probe the sea surface. Another application area, autonomous robot vehicle are used in the service sector. Robots provide intelligent services which further enrich the life quality of many people.

1.8.2 Robot Navigation

Robot navigation assigns autonomous robot to move from one site to another site. The robots should make appropriate decision to where it is going or how to get the desired location or goal.

Difficulties are obtain during the navigation process includes some consequences that affect the navigation process are adversity in tracking and recognizing objects, limits of computational power, collision avoidance and complexities engage in usage of knowledge produce by the environment.

- **Computational Power:** The CPU performance is not fast and thus do not provide enough power to perform the tasks. Real time image processing, computer vision are example required for computational power.
- **Objects Recognition:** Robots navigate to recognize the structures like objects and location in order to achieve their tasks. Image processing will boost more computational power if the structures are not known or whether the structures are known, recognition of objects and location is uncertain.
- **Collision Avoidance:** During the navigation process, robot should avoid colliding with objects in their environment. In complex problem, where multiple objects are moving, collision avoidance might be problematic. Robot uses certain path planning techniques to overcome collision.
- **Sensors:** Robots will use sensors to give information about the environment. For odor localizing, robots will be equipped by chemical sensors to investigate the tasks. Therefore, robots with olfaction technology are widely used in application areas including safety, environmental inspection and security purposes.

For localizing odor sources, robots will be equipped with gas sensing devices, detecting multiple odor sources and would respond quickly to an analyte gas. Therefore number of related odor source localizing will be performed by the mobile olfactory robot. This navigating task includes navigation for a large area or generating concentration outline of an area. It is significantly used in security and safety purposes. Studies focussed on developing research platforms for mobile robots are not proposed for real-world operation. Requirements from robots that can be used in swarm robotic system include:

- **Sensing and signalling:** Leaving some marks in the surrounding, robots will be able to sense them.
- **Communication:** Robots should support wireless communication channel and Programming the robots would be an immense time saver.
- **Physical Interaction:** Robots should be skilled enough to interact with each other and the surrounding.
- **Power:** Robots must have a long battery life.
- **Cost:** It should be cheap, sold at least in groups of ten.
- **Size:** In swarm robotic systems, size matters when experimenting with the systems.

CHAPTER 2
REVIEW OF LITERATURE

Yu-jun Zheng *et al.,* (2014) have proposed a paper based on population classification using MPSO tactic. The data are extracted from real world fire evacuation that occurred in china. The data are into different category with multiple attributes. As the real world data consists of mostly noisy, incomplete and inconsistency data. Data are then classified using the classification method IF-THEN rule and this rule is combined with multi-objective particle swarm optimization for optimization of the problem. Precision and recall measures are the two main roles of classification rules. Also to maintain the variety of the swarm's effective classification rules and broad learning strategy is defined. Paper shows the comparative analysis between the proposed method and other methods such as MOPSO, MOPSO-A, EMOGA, MOANT are evaluated and results show that the proposed method outperforms all the other method. Thus, the proposed method has been successfully defined in this paper.

Gurpreet Kaur and Rajdavinder Singh (2014) have proposed a paper which is based on solving a real world optimization. Techniques such as contourlet transform and firefly algorithm are used to improve the quality of the image. Also numbers of parameters are evaluated for the better working of this technique. For the improved contourlet transform, firefly algorithm is performed for proper sharpening of the ultra sound images. Thus result indicates that firefly algorithm out performs the existing solutions with its proper sharpening of the image.

Jianhua Zhang, Dunwei Gong and Yong Zhang (2013) presented this paper for localizing multiple odor sources by multi-robot cooperation method based on niching particle swarm optimization. In this paper a robot is regarded as the particle. The robots being distributed randomly in the search space, finding a plume during the search. Once the plume is found the robot and its neighbour will form a niche. The robots in the niche will begin to trace the plume using PSO algorithm. Update the position and velocity of each particle and the iteration continues until the maximum search time has reached or has found all the odor sources. Velocity and recovery time are considered when updating the position of the odor source. Also merging niches tactic is presented in order to prevent more than one niches searching simultaneously for the same odor source. Thus, this proposed paper results in finding the high success rate within short time consumption for localizing all odor sources by the robot.

Siti Nurmaini, Bambang Tutuko and Aulia Rahman (2013) proposed paper for swarm robots to detect odor sources in an indoor environment. This paper focuses on Interval Type-2 Fuzzy Logic and Particle Swarm Optimization algorithm for olfaction approaches. From proposed paper, input is taken as the location of the neighbouring robots and the output depends on the confined data gathered. Consistency and effectiveness is calculated in this proposed paper.

Xin-She Yang and Xingshi He (2013) purposed paper on advances and applications of firefly algorithm. Firefly algorithm is a multi-modal optimization and this paper focuses on different advantages and application that a firefly algorithm is efficient than other meta-heuristics algorithm. Firefly algorithm is inspired from the activities of fireflies. Fireflies produce a luminescence light. All firefly are unisex that one firefly will get attracted to any other firefly. Brightness is proportional to the landscape of the objective function. Brightness of a firefly decreases as the distance increases. Attractiveness is directly proportional to the brightness and that the less bright firefly will get attracted to the brighter firefly. Also this algorithm is evaluated in applications like digital image compression, in multi-modal problems and outperforms PSO to obtain the optimum solutions. Efficiently used in clustering and classification methods. An advantage of firefly which makes this algorithm very efficient is based on the subdivision property and the ability to obtain multimodality function. Because of this advantages firefly can find optimal solutions simultaneously even in a large set of population. This algorithm is also applicable for irregular search strategy with two phases includes: slow phase and the fast phase, slow phase is for local search techniques and fast phase deals with the exploitation technique. Thus, in intermittent search strategy firefly algorithm outperforms other algorithms. Further studies requires meta-heuristics algorithm in the area of combinatorial optimization.

Zahra Honarpisheh and Karim Faez (2013) presented a paper based on the dorsal hand vein recognition based on firefly algorithm. It uses Pattern recognition which acts as another type of biometrics. The author first analysed the blood vessels on back of the hands. Pre-processing and feature selection is identified using clustering algorithms. Two methods far infrared (FIR) and near infrared (NIR) were used to record dorsal hand vein images. This paper focussed on the identification of the pattern between crossing vessels and their matching place. Results obtained are the high speed in pattern recognition and less computation. Proposed paper focussed on hand vein because it contains high error coefficient and low recognition rate. Recognition based firefly algorithm could also happen in face and retina through LDA and PCA methods. Charged-couple device cameras prepared with LED lamps are used to set image of veins.

Qiang Lee and Qing-Long Han (2011) in their paper deals with localizing odor sources in multiple-robot system. Proposed paper defines certain measures, location of the odor source is calculated using a particle filter with the position obtained, where the leader robot will have a movement direction to trace the plume, thirdly for the movement adjustment whether in a circular or parallel motion, a decision making control laws have been designed and finally performance capabilities for odor source localizing is found.

Ali Marjovi *et al.*, (2010) in their research paper robot swarm olfactory based navigation method to discover odor sources in an unknown environment. Each robot has a localizing system and distance between robots is measured by the wireless network. The capabilities of smell, communication and to avoid obstacles are strategy for this algorithm. Proposed paper defines at least of three robots which acts as a static dimension beacons while other robots swarm around to find the odor sources. Next the robots will switch and some may remain static while others may find the odor sources. Stages that a robot can take in this method: scatter, aggregate, beacon and odor source localization. Robots were fit with infrared and sonar sensors. Sensors are used for obstacles avoidance. Each robot uses Received Signal Strength Indicator (RSSI) of Zig Bee communication module for the distance measurement and particle filter for its current position.

Qiang Lu, Shi-dong Liu and Xue-na Qiu (2010) paper deals with localizing odor source using multi-robot systems. Two layers of distributed architecture is proposed: architecture layer and control layer. Evolutionary algorithms are employed in AI to gather information from other robots via transmission networks. On the other hand, a consensus algorithm is employed in control layer to state the change of robot from the present to the new derived state. The new derived state is measured by evolutionary algorithm. Different environments are calculated during performance include: small wind, medium wind and large wind respectively. In this paper effectiveness is measured with the two layers in a distributed architecture.

W. Jatmiko *et al*., (2009) proposed a paper to track a plume in a local gradient of chemical concentration using Modified Particle Swarm Optimization (MPSO) with the direction of the wind velocity. A niching feature is elected for solving the multi-peak and multi-source problem. During the parallel search of the odor source, each subgroup of the robot will search for the odor sources. High chances of localizing an odor source by more than one subgroup. Thus, it is not efficient as other subgroup will locate other source. A ranged method is employed for the subgroup problem, as to increase the searching performance. For the balancing moment, friction of the robot we used Open Dynamics Engine Library.

Thomas Lochmatter and Alcherio Mertinoli (2009) have proposed a paper on theoretical analysis of three bio-inspired plume tracking algorithms. The algorithms proposed in this paper are all bio-inspired which includes the casting, surge-spiral and surge-cast algorithm. It is based on the wind sensor which is to measure the wind direction. Based only on the plume tracking actions, the robot starts tracking in the plume and neglects if it cross the desired area. Basically this paper is based on wind flow where the robot moves upwind, cross-wind following the wind direction with the combination of this used algorithms. Also, Bayes inference figures means success rate and distribution of the distance overhead. This paper represents a parallel analysis over existing real-robot and simulation analysis.

Hiroshi Ishida *et al.,* (2005) proposed a paper on factors overcoming the limitations of plume-tracking robot using a transient response-based algorithm. The robots are equipped with gas sensors to track odor plume and determine its source. Basically the paper is based on breaking the limitation of slow response of gas sensors by locating onsets of gas sensor response and with the change in the output, therefore recovery of the plume-tracking starts by showing off with a faster performance. The robot speed increases accelerating in the development of plume tracking over expanding sensor outputs, whereas degrades in success rate due to inadequate sensor outputs. Also recovery time (>60s) of a gas sensors are used. Based on the proposed algorithm which is transient-response based algorithm, which outperforms the limitations of gas sensors with a significant enhancement in the performance of the plume-tracking robot with an average speed of 13cm/s.

Borenstein *et al.,* (1989) proposed an approach for fast mobile robots to avoid collision and progress towards the target. Sometimes when mobile robots are in motion to achieving its target or its objective, there may be times when obstacles are detected and that collision might occur. Therefore, the proposed paper adopts a virtual force field approach for navigation and obstacle representation and allows robots to be in uninterrupted motion allowing vigorous behaviour of the robot. The proposed algorithm solves the local minimum trap problem for fast mobile robots.

CHAPTER 3
PRESENT WORK

In this chapter, we are going to present the problem of our research work, its objectives, the methodology that we used for our purposed approach and the introduction of the developed tool. In the 3.1 section we explain how we formulated our problem and what the approach we are going to use. In the 3.2 and 3.3 section the objectives and the methodology of the work done. In the methodology the flow of our work with the help of flow chart is explained.

3.1 PROBLEM FORMULATION

Firefly Algorithm is proposed to localize the odor sources in a niche. The proposed algorithm is a swarm technique which performs the behaviour of fireflies. The fireflies with the brightness and attractiveness outperform many swarm techniques. It has advantages over other techniques based on the attractiveness and multimodality. A firefly with less brightness will move towards the firefly with more brightness. But brightness will matter in terms of the distance between the fireflies.

With certain advantages, firefly algorithm has been proposed for localization of odor sources in a niche using multi-robot cooperation method. Also, brightness is associated with the odor sources and that the firefly with high fitness value will be able to localize the odor sources, because the brightness is more. Also multi-robot will get attracted to the brightest source for localization.

3.2 OBJECTIVES

The objective of the research is to localize odor sources by multi-robot cooperation method based on a niching firefly algorithm

- A niche formation with odor sources to be localized
- Multi-robots start searching for the odor source according to the proposed algorithm
- Several iterations are calculated to localize all the odor sources
- Calculating the fitness value and taken into comparison with the existing work
- Time consumption is calculated to localize all odor sources and compare this performance in order to achieve more efficiency results than the existing work.

3.3 METHODOLOGY

In the research paper proposed firefly algorithm to obtain the desired result i.e. to localized odor sources by multi-robot cooperation method in a niche. Firefly algorithm which is based on a nature-inspired behaviour of fireflies. It is a swarm intelligence algorithm for optimization. Firefly algorithm is also used for multi-modal functionality and it is observed to be more efficient than other algorithms.

3.3.1 Firefly Algorithm Description

The fireflies as agents moved around the search space to find the optimum solution. The parameters include its location and the light intensity or brightness. Initializing and computing the locations and brightness of the fireflies. Assigning the location of the brightness firefly in the population as the current global best. During the movement phase, update the location of the fireflies. The movement of one firefly is attracted towards the one with brighter firefly. It is given by the equation:

$$x_i = x_i + \beta_\circ e^{-\gamma r_{ij}^2}(x_i - x_j) + \propto \varepsilon_i \qquad (1)$$

Where $\beta_\circ$ and $e^{-\gamma r_{ij}^2}$ denotes attractiveness between two fireflies *i* and *j*

γ denotes light absorption coefficient and β is the attractiveness.

Distance between the two fireflies is the Cartesian distance. Cartesian distance between the two fireflies *i* and *j* at x_i and x_j is computed as:

$$r_{ij} = \|x_i - x_j\| = \sqrt{\sum_{k=1}^{d}(x_{i,k}} - x_{j,k})^2 \qquad (2)$$

With the attractiveness of the fireflies, at every iteration the locations and the global best solution have been updated. Iteration process continues until it reaches its maximum limit. The location of the best firefly as the global solution. Below shows the Firefly algorithm steps:

Step 1: Generate an objective function f(x), where

$x=(x_1,x_2,.......,x_d)$

Step 2: Generate an initial population of fireflies

$x_i(i= 1,2,....,n)$

Step 3: Determine light intensity I, associated with f(x)

Step 4: Define light absorption coefficient γ

Step 5: Number of iterations till a maximum time is obtained

while (*t* <MaxGeneration) **do**

for *i* = 1 : n all n fireflies **do**

for *j* = 1 : n all n fireflies **do**

Step 6: Comparison between two fireflies. Firefly with less bright will move to the brighter one

if (I_j>I_i) move firefly *i* towards *j*

end if

Step 7: Attractiveness varies with distance *r* via exp [-γr^2]

Step 8: Evaluate new solutions and update light intensity

end for *j*

end for *i*

Step 9: Rank them and determine the current best

Step 10: End

3.3.2 Firefly Algorithm Rules

1. All fireflies are unisexual so that one firefly will be attracted to any other firefly.
2. Attractiveness is proportional to brightness, such that firefly with less brightness will get attracted to the brighter firefly.
3. Increase in distance will decrease the brightness of the firefly.
4. Brightness of a firefly is associated with the objective function.

3.3.3 Proposed Algorithm

Algorithm starts by placing the fireflies randomly in the search space so that it is well dispersed. Initialized the location of the fireflies and also computing the brightness of the fireflies for maximization problem. The brightness is proportional to the value of the objective function. In the movement phase the bright one get attracted towards the brighter firefly. The attractiveness between the two fireflies is denoted by the Cartesian distance. The brightness will vary in increased of the distance. Attractiveness is directly proportional to the brightness of the firefly. The objective function is associated to the brightness of a firefly. Each iteration will result in the updating of the position or location of the firefly. Repeat iteration until it reaches its maximum limit and returns the position of the best firefly as the global solution.

Steps for working of firefly can be described as follows:

1. Niche formation
2. Search for odor in the niche
3. Initialize the position of the firefly and generate light intensity
4. Calculate maximum iteration to localize all odor source.
5. Calculate the light intensity for each odor source.
6. Update the current position
7. Calculate the distance between the particle and the odor source
8. Calculate collision rate
9. Select the optimal distance between two particles. The optimal position is obtained.
10. Iterate step 6 to step 9 until the position of all odor sources are localized.

Below shows the flowchart working of firefly algorithm in a multi robot cooperation method for localizing odor sources. In a particular niche, robot will search for the odor sources. The brightness is associated with the odor source. In the movement phase, computing the position changes and the light intensity will vary according to the fitness value obtained. Calculating the distances and updating the position of the particle and finding the best fitness value as the optimal position. Fitness value near to the target will have the brightest and that the firefly with less brightness will be attracted to the brightest ones. Thus, brightness is associated with the odor source, the robot regarding as a firefly will search for odor source and that the firefly with high fitness value and the brightness of the odor sources will localize the source. Also, robots with brightest one will be attracted to the odor source. Measures are obtained to avoid collisions of the particle during the searching process. Several iteration processes will be calculated with fitness and time consumption in order to localize all the odor sources, till all the odor sources have been localized. The number of iteration process will continue till when all the odor sources are localized by the robot, calculating the average fitness value and time consumption taken by the proposed method.

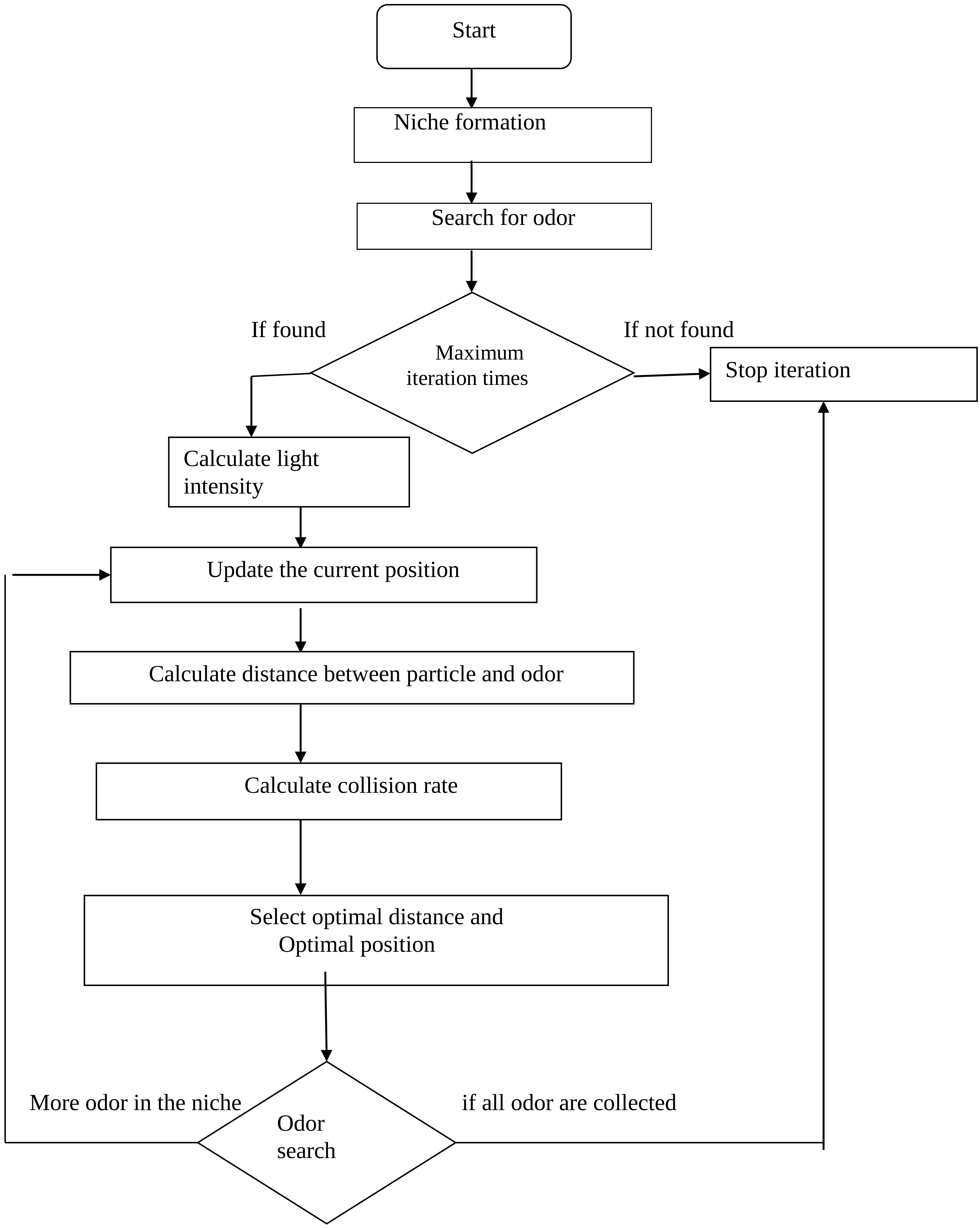

Fig: 3.1 Niching Firefly Algorithm

The proposed algorithm for localizing the odor sources using a multi-robot method will perform experiments on matlab. Here represents a few general information about matlab.

3.4 MATLAB

Matlab stands for "**Matrix Laboratory**", a multi-paradigm numerical computing environment. Matlab is a high level language. It allows Matrix manipulation, implementation of algorithms, user interfaces and interfaces with languages namely, C, C++, Java and Phyton. Matlab was originally designed for solving linear algebra type problems using matrix. Matlab system has following these parts:

- Mathematical function library
- Desktop tools and development environment
- The language
- Graphics
- External interfaces

3.4.1 FEATURES OF MATLAB

- High level language.
- Provide interactive tools for solving problems.
- 2-D and 3-D graphics functions for analysing data.
- Environment for managing code, files and data.
- Provide functions for integrating the MATLAB based algorithms with external application.
- Maintainability and maximization performance.

3.4.2 STANDARDS WINDOWS IN MATLAB

- **Command Window:** For type and execute commands.
- **Workspace Window:** To create variables and store in memory during a MATLAB session, edit variables by double-clicking and opening an array editor, whereby loading variables from files and to clear variables.
- **Current Directory Window:** This shows MATLAB files and current directory in current folder, also can change folders and load files.
- **History Window:** This shows previously executed commands. Re-execute commands by double-clicking.

3.4.3 MATLAB help

- Help option is present on the right side of the top window.
- It is a dynamic means for learning MATLAB.
- Other than theoretical background, also shows demos for implementation.
- Searching any command in the search box.
- Commands searched are explained with examples.

3.4.4 MATLAB Input and Output Commands

- **disp: -** Displays contents of a string or an array
- **input:** - Waits for input and displays prompt
- **fscanf:** - Read formatting data
- **fprintf:** - formats writes to file

3.4.5 Advantages and Disadvantages of Matlab

- **Advantages**
  - Ease of use
  - Matlab is a platform independent
  - Predefined functions
  - Plotting
- **Disadvantages**
  - It is expensive
  - It can be also slow

3.4.6 Plotting Commands

- **axis:** setting the limits of axis
- **plot:** xy plot
- **xlabel:** in x-axis adds text label
- **ylabel:** in y-axis adds text table
- **bar:** to create bar chart
- **contour:** to create contour plot

CHAPTER 4
RESULTS AND DISCUSSIONS

In this chapter we have presented various results based on the proposed method. Various graphs are generated to give the fitness value of the robot, short time consumption with an efficient result using firefly algorithm. With several performances the result is clear that the proposed algorithm is better than the previously used techniques. The results are shown with several numbers of input values. The results are as shown:

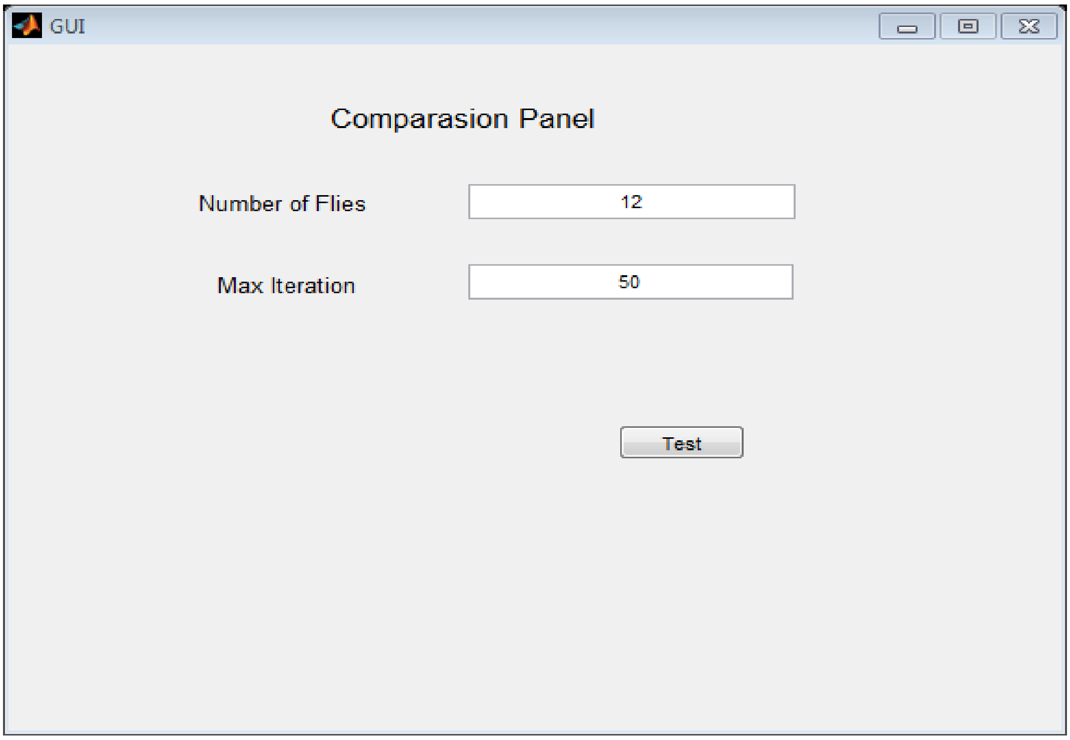

Fig: 4.1 A simple GUI

A simple GUI to enter the number of robots and specify maximum iteration times to perform the localization of odor sources. In order to give a clear understanding of the output, the number of iterations should be large enough.

In our experiment we have used flies as the robot to localize odor sources in a particular range. The limits for localizing odor sources is in a fixed range of 5 x 5 m^2 are generated by matlab. The odor sources are defined in this particular range where the number of fireflies will search for the odor sources. This figure shows the localized odor sources in a defined range.

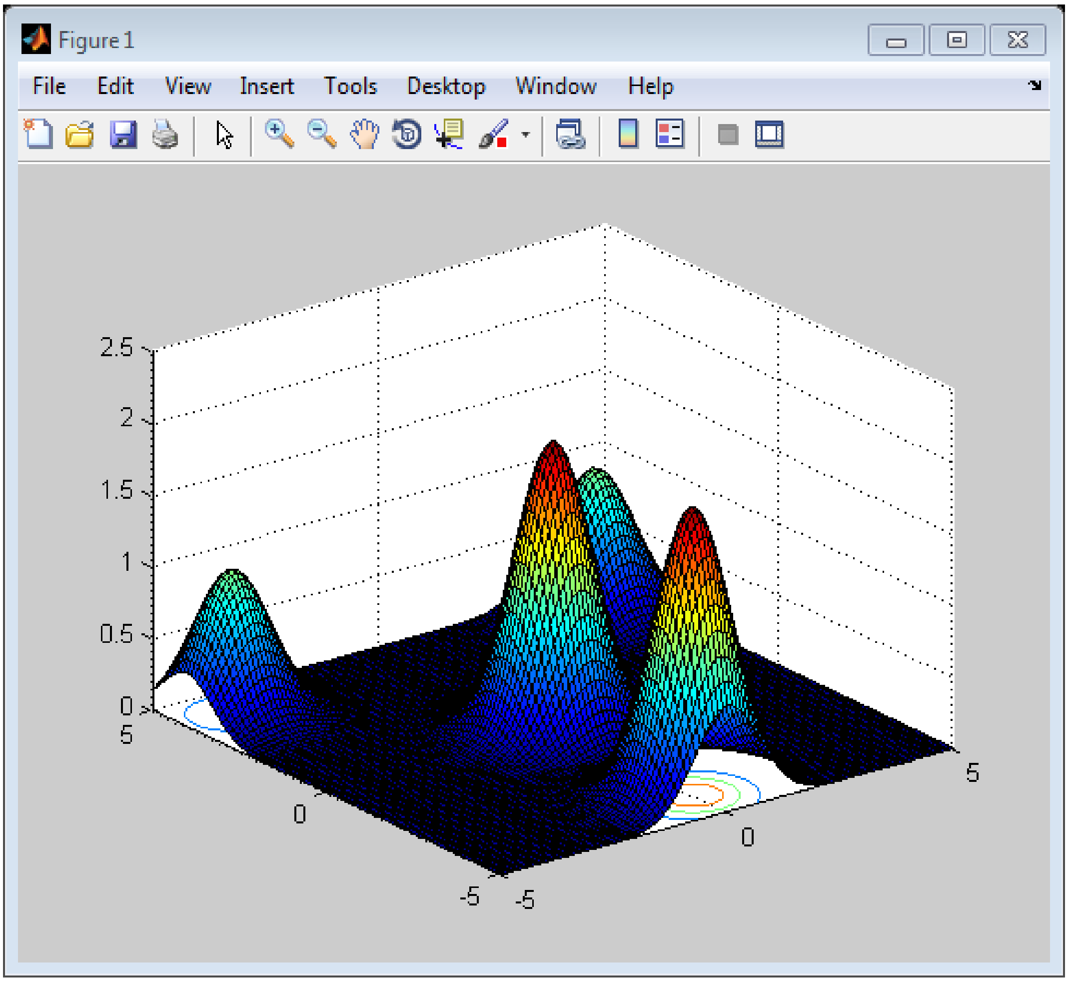

Fig: 4.2 3-D dimensional graphs

This figure plots the area where the odor sources are defined and shows the brightest area as odor localized by the robots. As the proposed algorithm, light intensity is defined for each firefly and the corresponding fitness value, the flies which will be acting as a robot will search for the odor source, also odor sources is associated to the brightness. At the movement phase, firefly with less brightness will get attracted to the brightest ones. With the number of iterations, movement of firefly will result in the fitness value; flies nearest to the target will have the best fitness value than the others. As a result, more brightness the firefly will be and that the firefly with less brightness will get attracted to the brightest one. Thus, flies will localize all the odor sources with accordance to the brightness.

RESULTS AND COMPARISON OF FIREFLY ALGORITHM WITH PARTICLE SWARM OPTIMIZATION WITH DIFFERENT INPUT VALUES IN MATLAB

In this section we presented the results with different inputs and the maximum number of iterations to localize all odor sources by multi-robots method using the proposed algorithm. Multiple input and iterations are presented to achieve the goals. The experiment will be evaluated for multiple iterations and the short time consumption of an algorithm till all the odor sources are localized by the robot.

Below figure will show the multi-robots searching for the odor sources with given number of flies and iterations. The process continues searching for the odor sources.

- No of flies = 10

  Maximum iteration times = 50

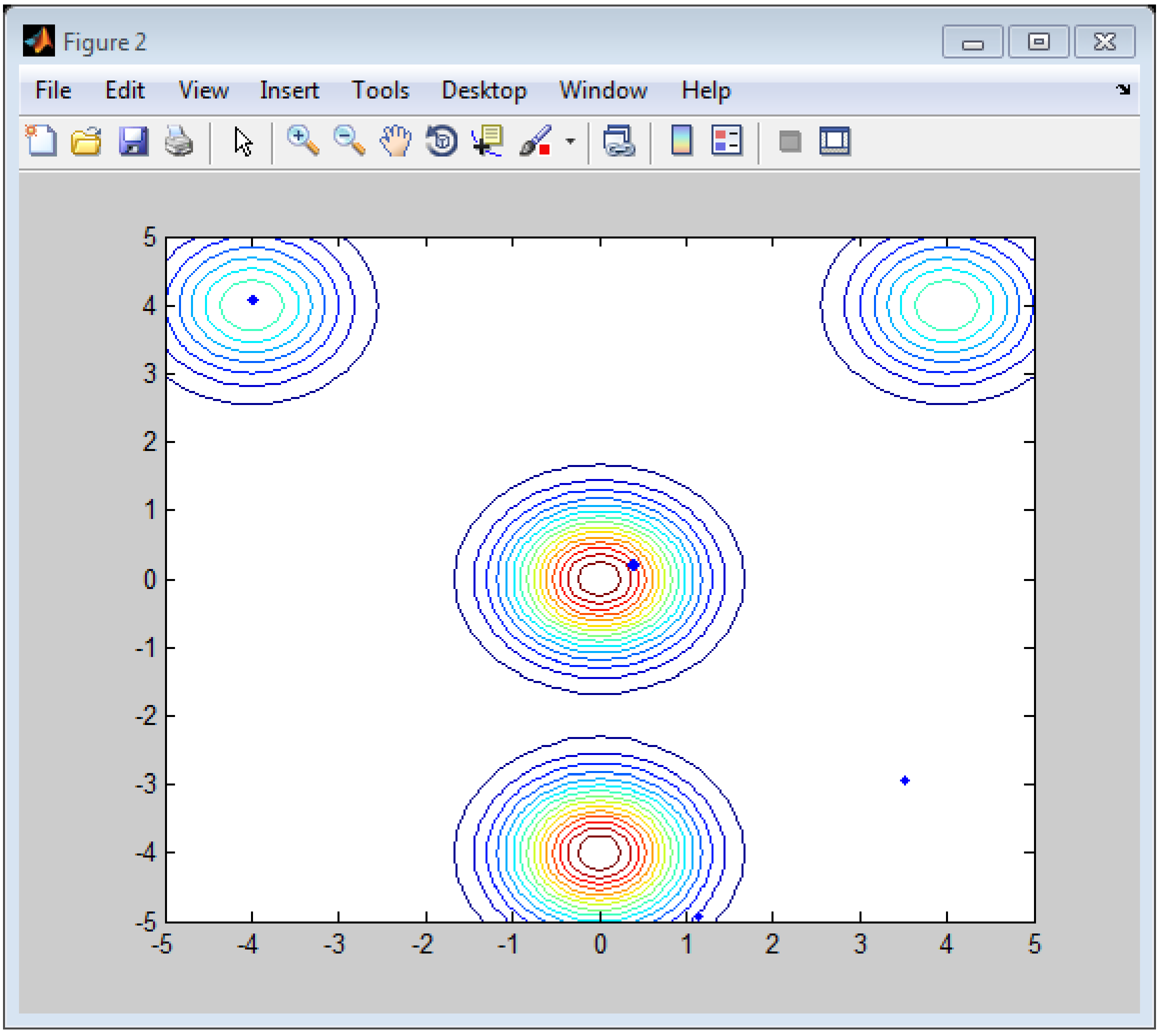

Fig: 4.3 Odor localization

Below figure is showing the intensity of the given flies. The first two flies has very less brightness, where 3 and 4 have intensity brighter than 1 and 2 and the remaining flies has almost equal intensity of brightness. As in the localization figure, flies with less brightness could not localize the odor and are out of the odor ring whereas rest of the flies has localize according to its brightness level.

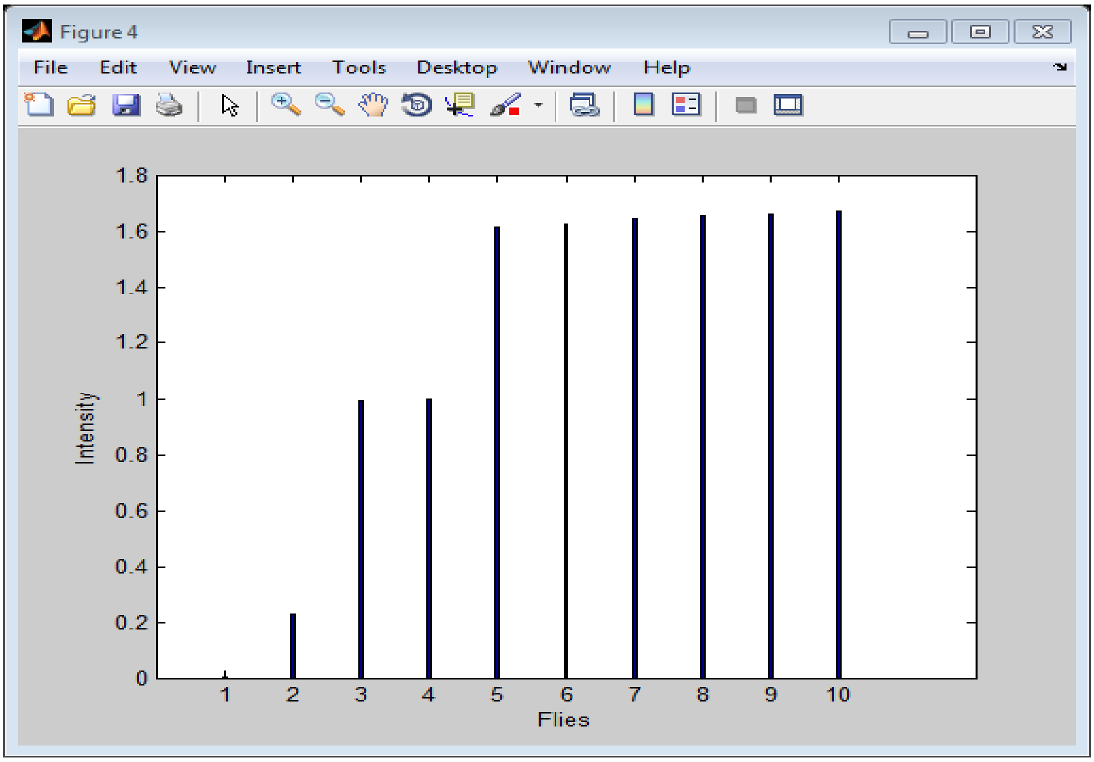

Fig: 4.4 Intensity of the flies

The fitness value will determine according to the given flies and maximum iteration times. Maximum iteration will result in giving the best fitness value of proposed algorithm with the existing algorithm.

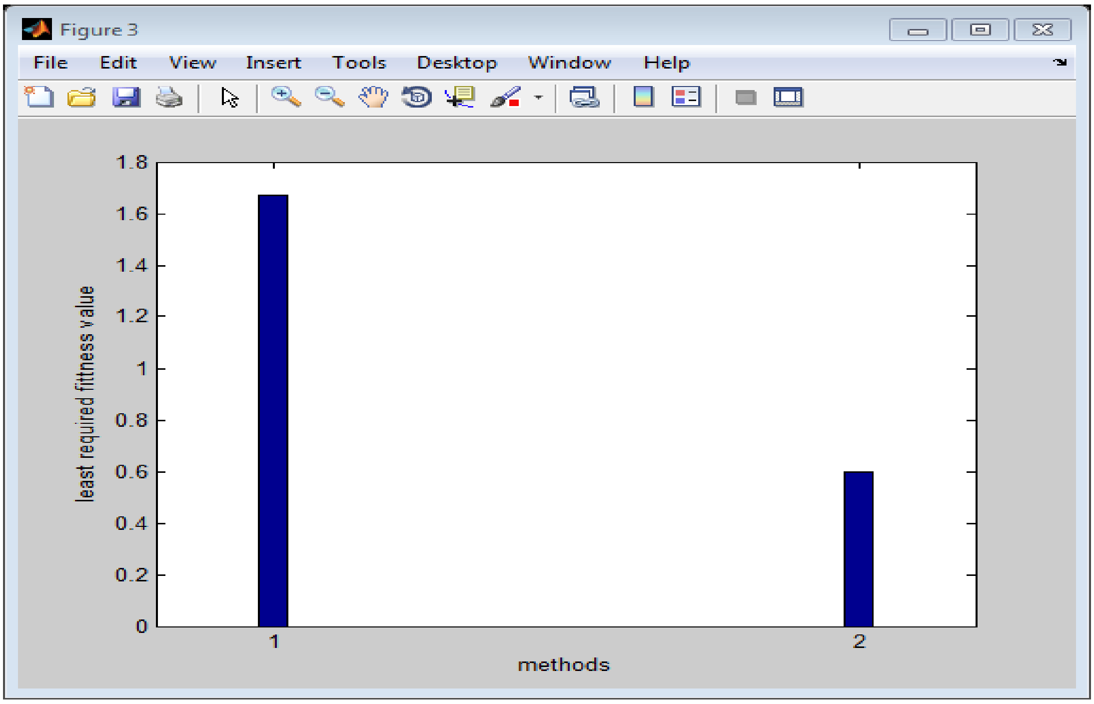

Fig: 4.5 Fitness value

Figure shows the time consumption of proposed algorithm with the previously used algorithm during the localization of odor sources.

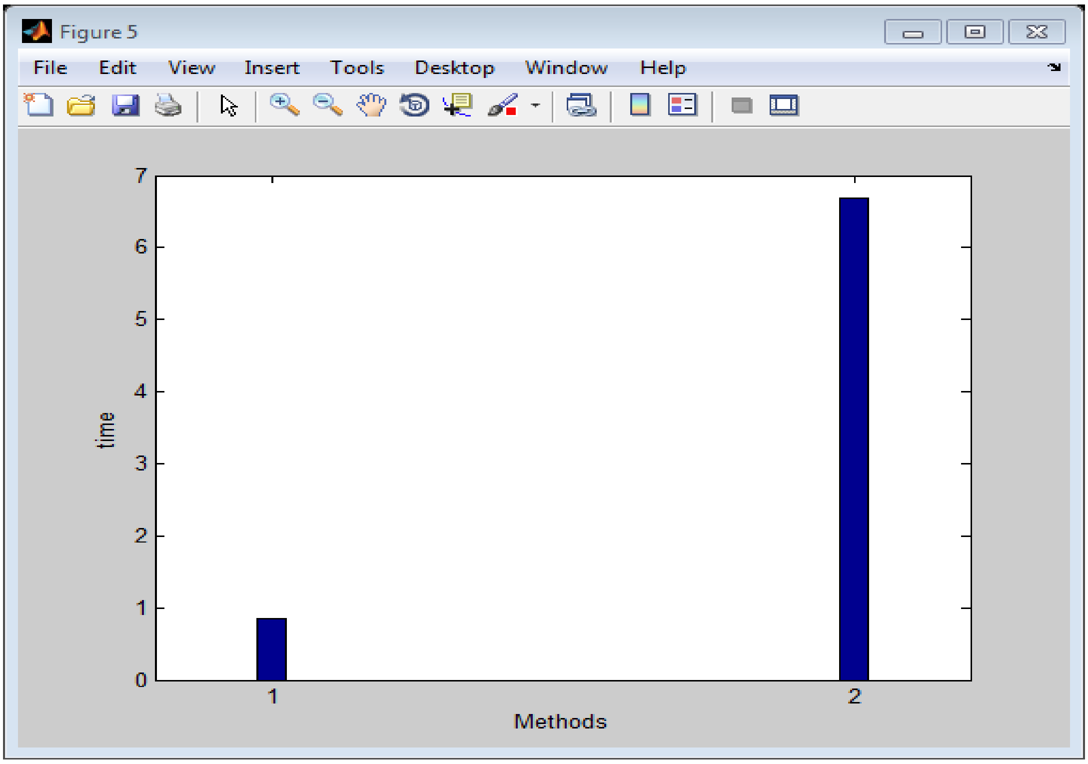

Fig: 4.6 Time consumption

Below figure shows the collision rate comparison that occurs while searching for the odor sources with respect to the given number of flies and the iteration times.

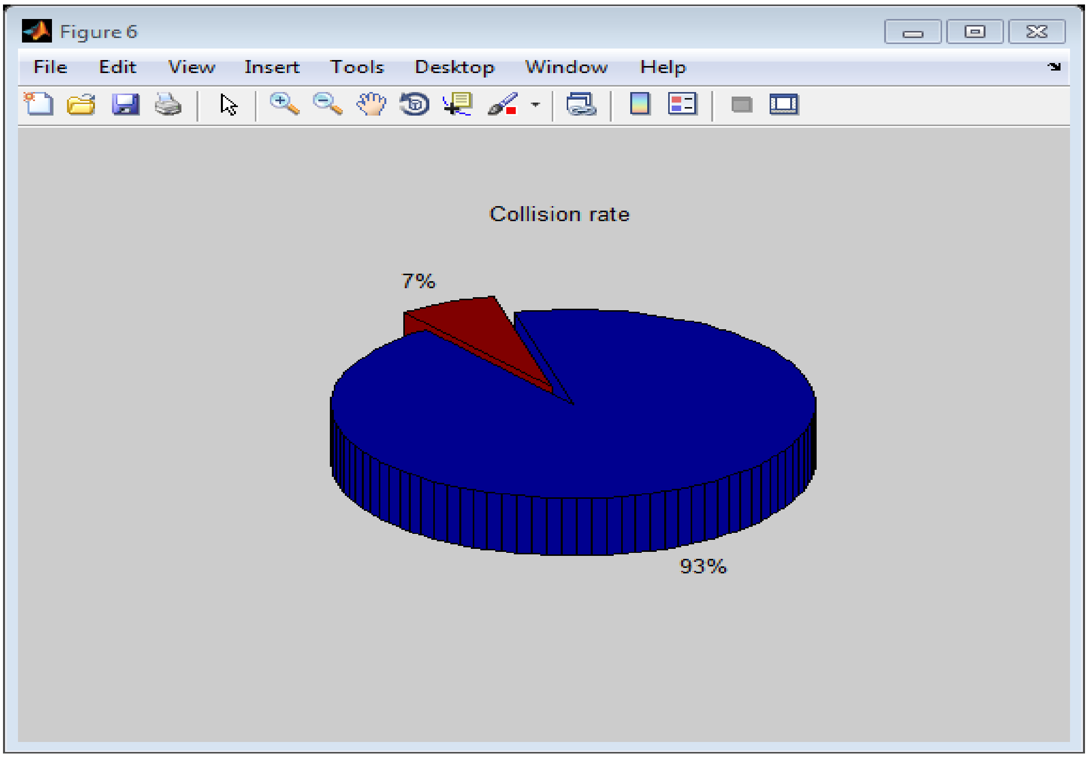

Fig: 4.7 Collision rate

➢ No of flies = 15

Maximum iteration times = 60

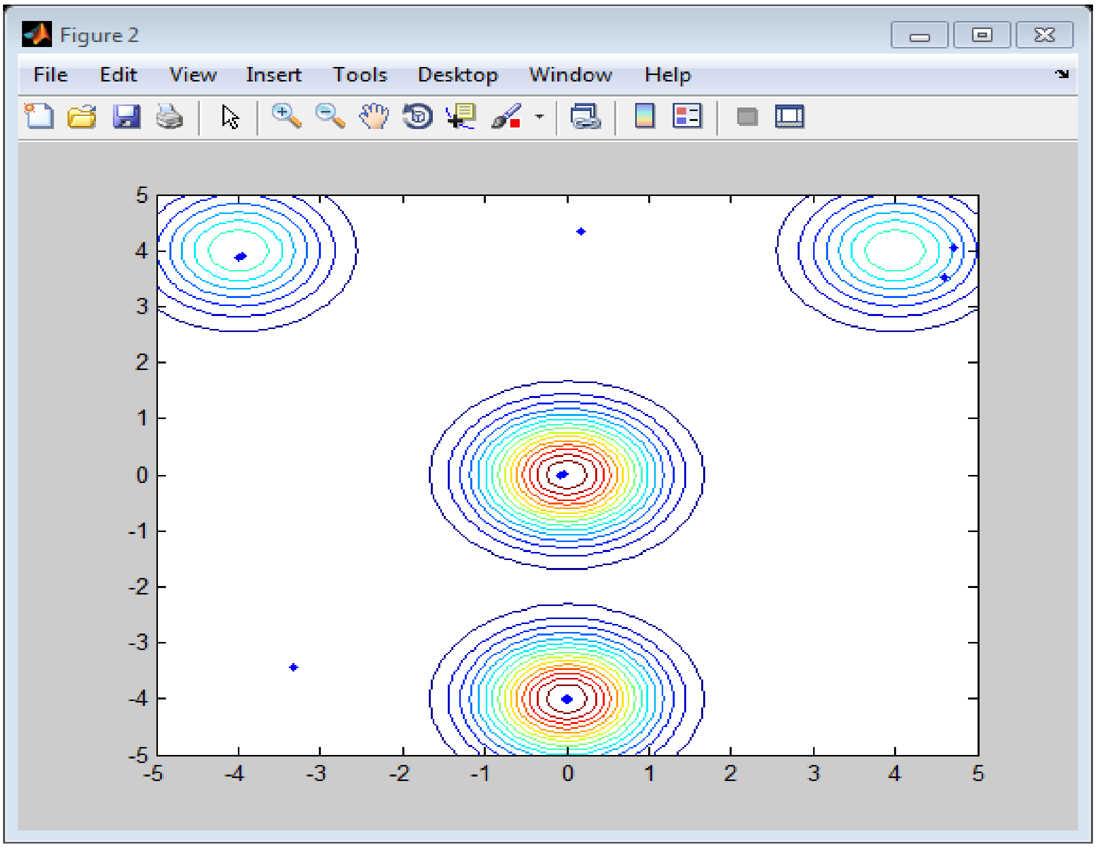

Fig: 4.8 Odor localization

Below figure shows the different intensities of the given flies for localization.

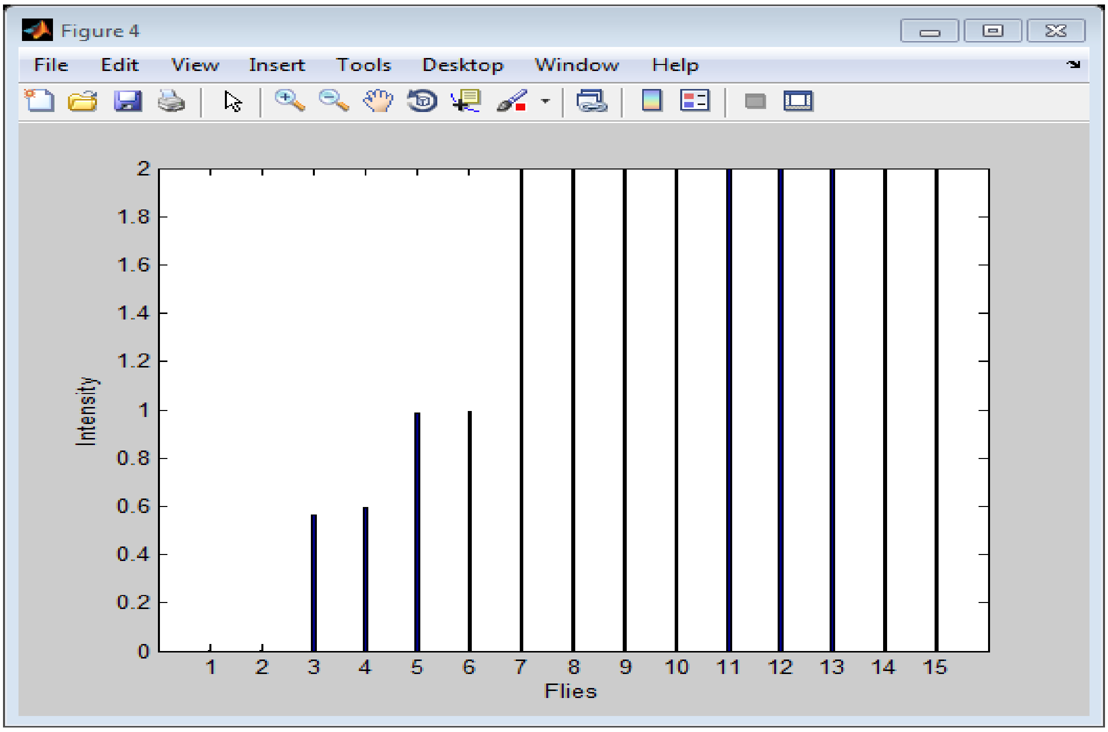

Fig: 4.9 Intensity of the flies

From the above figure 4.8 and figure 4.9, flies localized odor based on their intensities. Flies 1 and 2 have zero intensity, 3 and 4, 5 and 6 has the same level of brightness and the rest of the flies have more intensities of brightness. In the localization phase, no bright intensity will not localize and are out of the odor ring, and the corresponding flies will localize odor based on their brightness intensities.

The figure below will show the fitness value comparison with respect to the given number of flies and maximum iteration times. Output will be clear with more number of iteration times and the corresponding fitness value of the proposed algorithm.

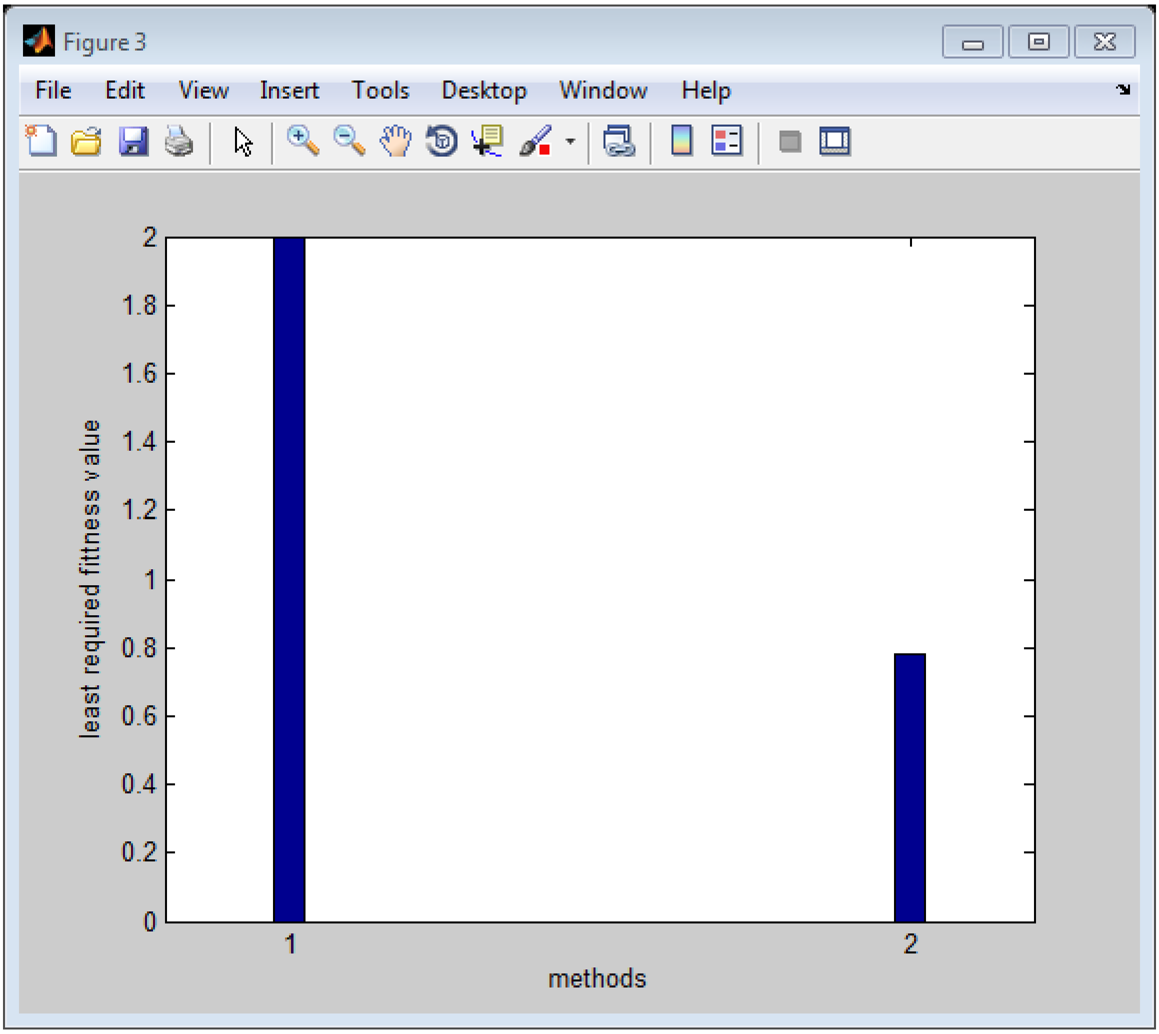

Fig: 4.10 fitness value

Figure shows the time consumption of proposed algorithm with the previously used algorithm during the localization of odor sources. Time accuracy is calculated with respect to odor localization in a niche using the proposed method in comparison with the existing work.

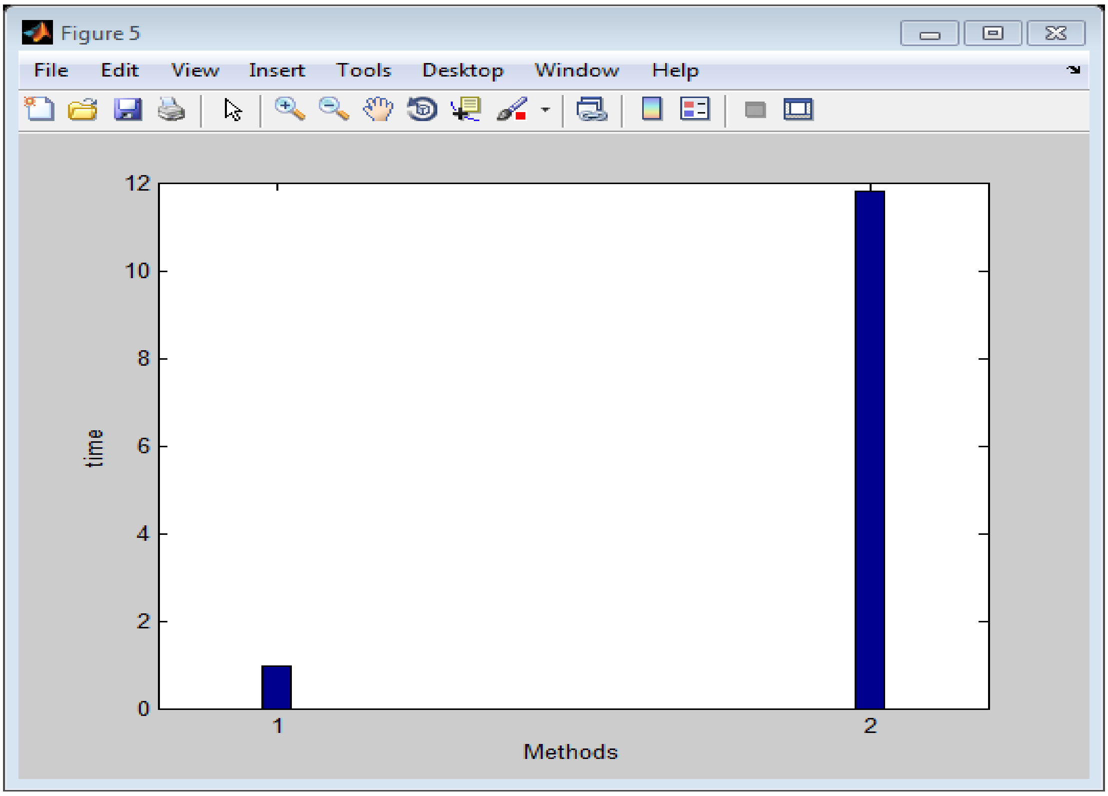

Fig: 4.11 Time consumption

Below figure shows the collision rate comparison that occurs while searching for the odor sources with respect to the given number of flies and the iteration times.

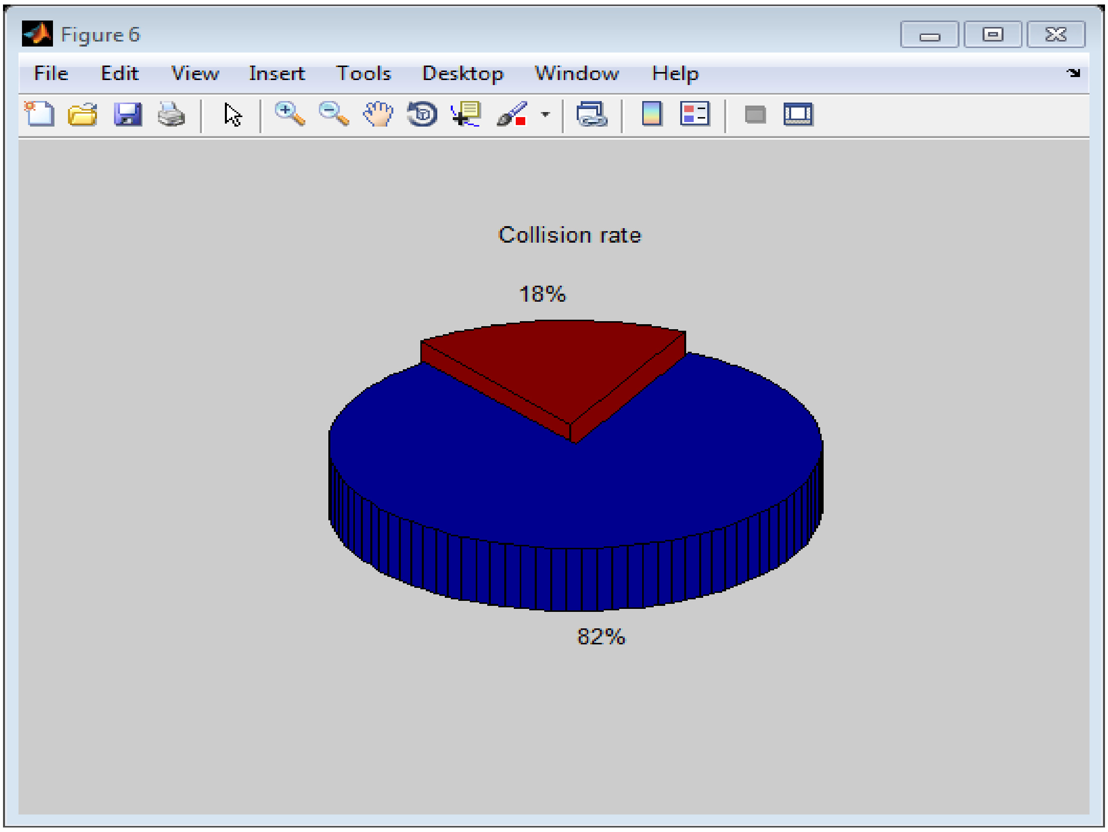

Fig: 4.12 Collision rate

This experiment shows the comparison of firefly algorithm with the previously used algorithm in terms of fitness value, time consumption. Here, observations for several numbers of cases with different input values are specified where all odor sources are localized. Though all shows the localization of odor sources but it will vary with different fitness value and time consumption.

Case: 1

No of flies = 12

Maximum iteration = 50

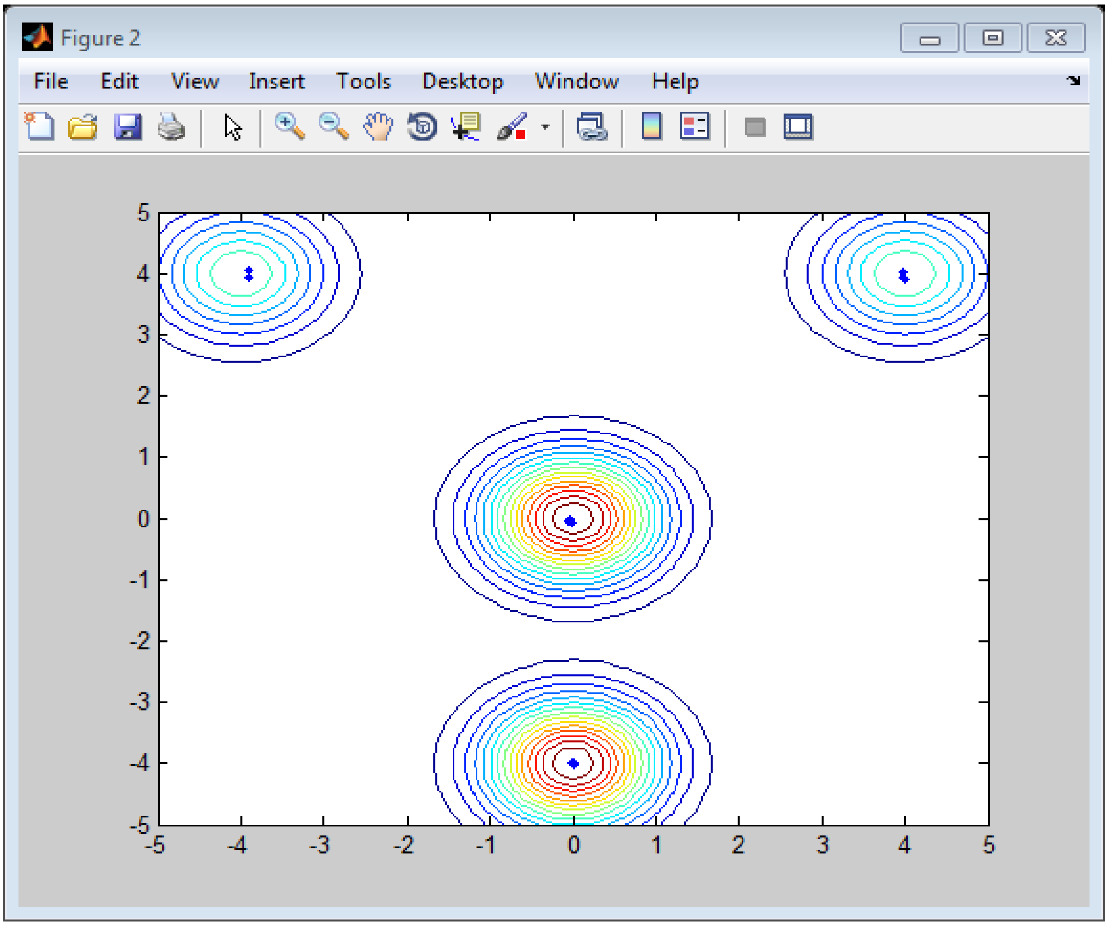

Fig: 4.13 Odor localization using FA

In the above figure, all the flies have localised the odor sources according to the brightness or intensity of the flies. Flies 1, 2, 3, 4 and 5 have almost the same intensity of light and are localized accordingly, whereas the remaining flies have brighter intensity of light. Because brightness will matter according to the fitness value and also features of proposed algorithm is being performed.

Below shown figure is showing the fitness value comparison of FA with PSO

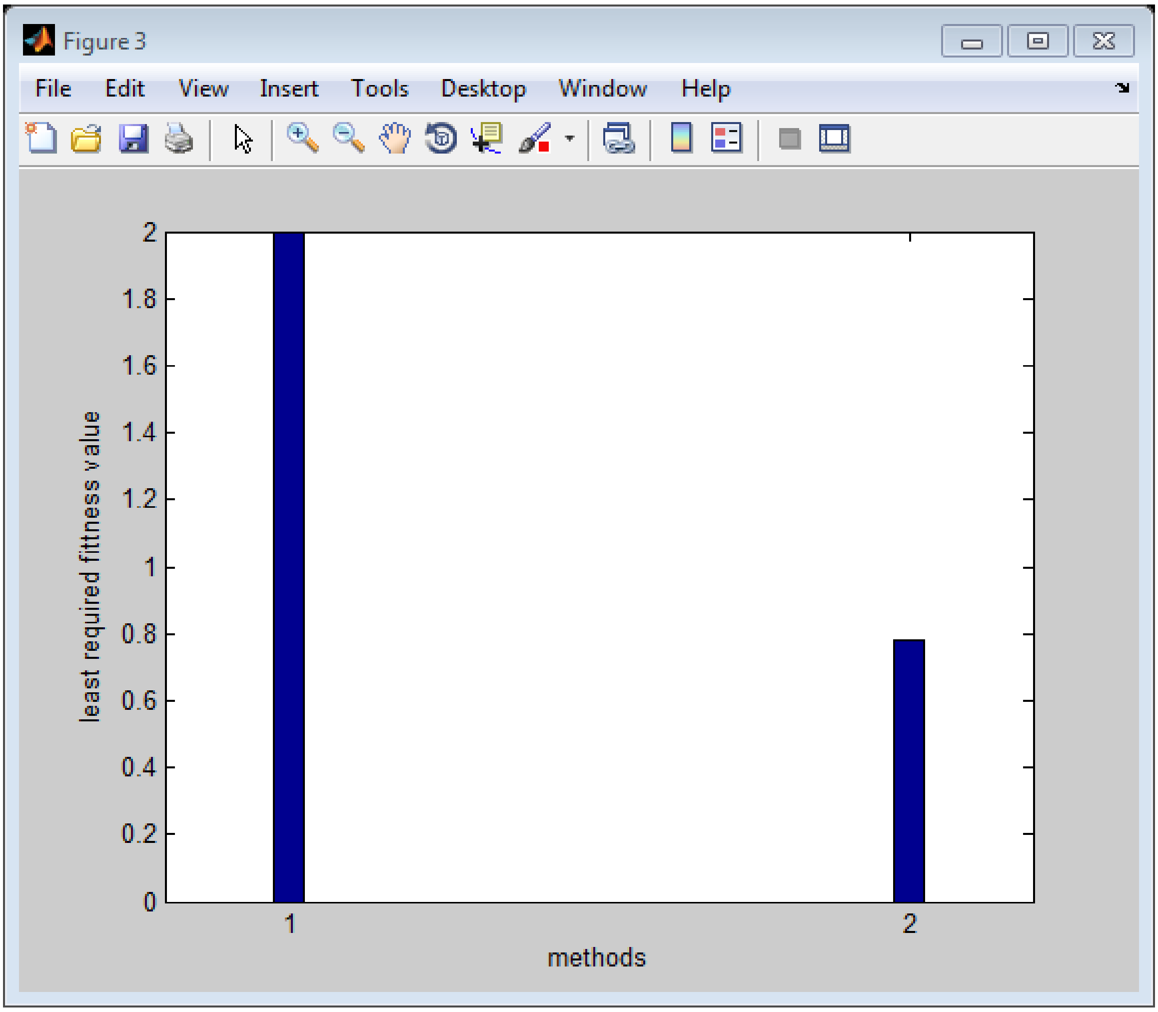

Fig: 4.14 Fitness value

Below shown figure is showing the intensity of the given flies taken to localize odor source. In the proposed method, brightness and attractiveness are quite an important factor where a fly with less bright will get attracted to the fly with more brightness. The given number of flies in this case will have different intensity and also brightness will increase with iteration times because near to the target or the odor source, more the intensity increases.

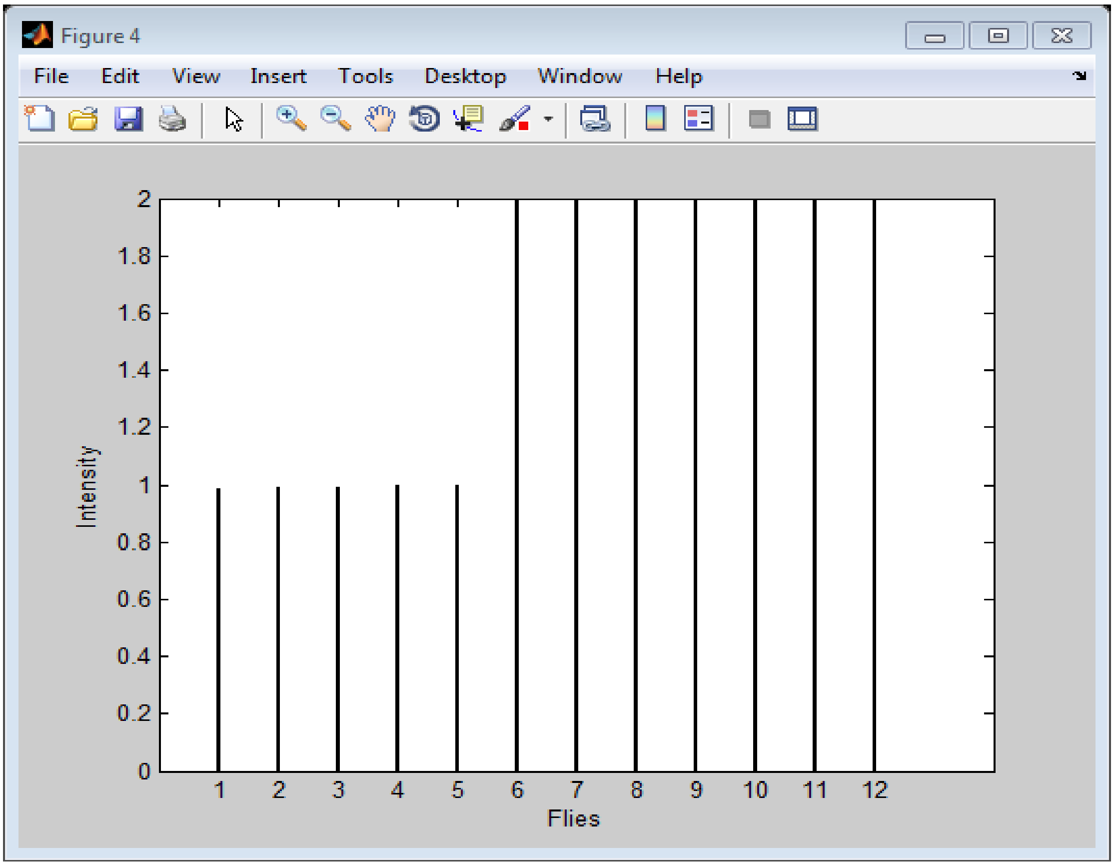

Fig: 4.15 Intensity of the flies

Below shown figure is showing the time consumption of FA with PSO

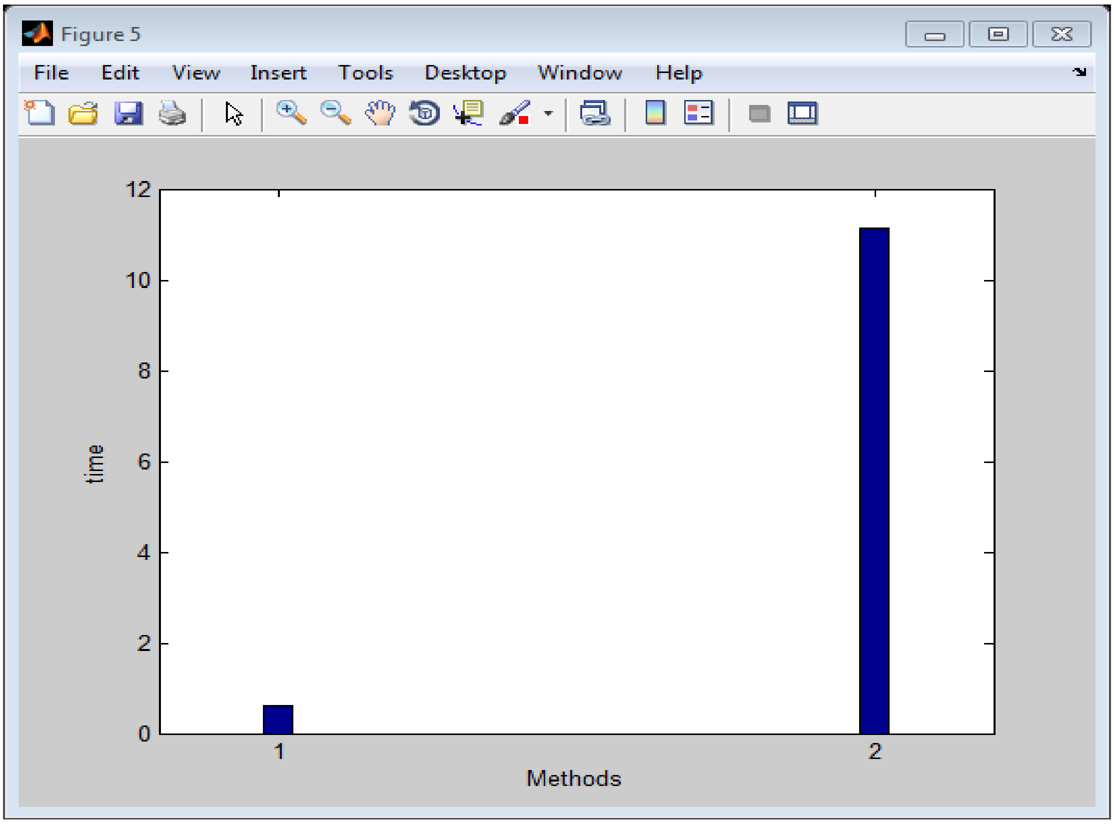

Fig: 4.16 Time Consumption

Below figure shows the collision rate comparison that occurs while searching for the odor sources with respect to the given number of flies and the iteration times.

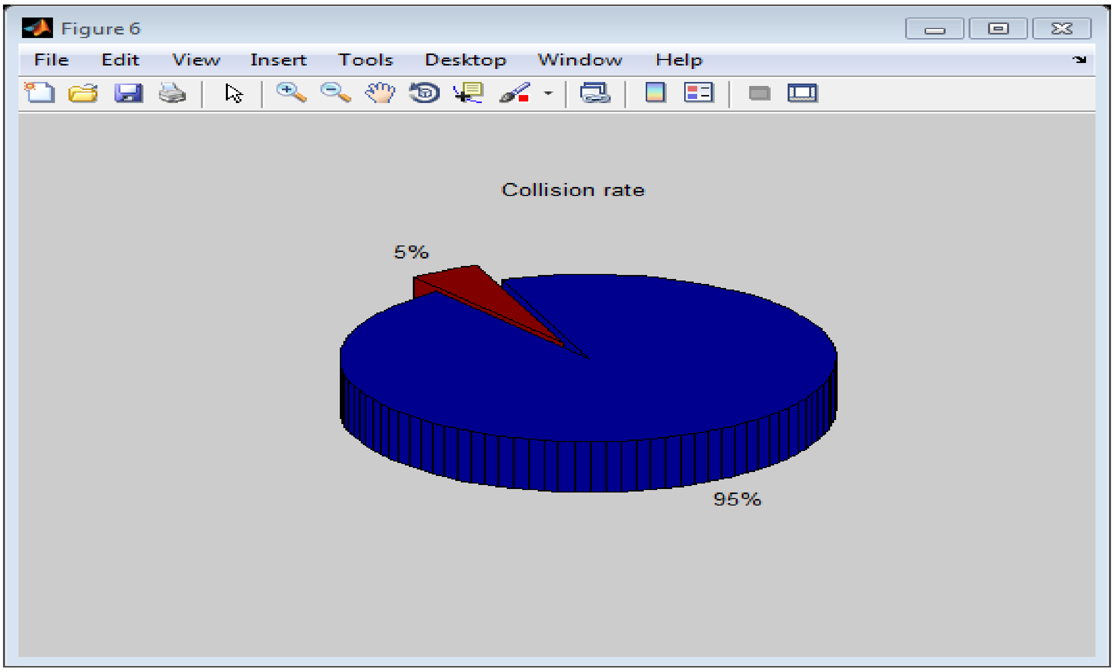

Fig: 4.17 Collision rate

<u>Case: 2</u>

No of flies = 25

Maximum iteration = 80

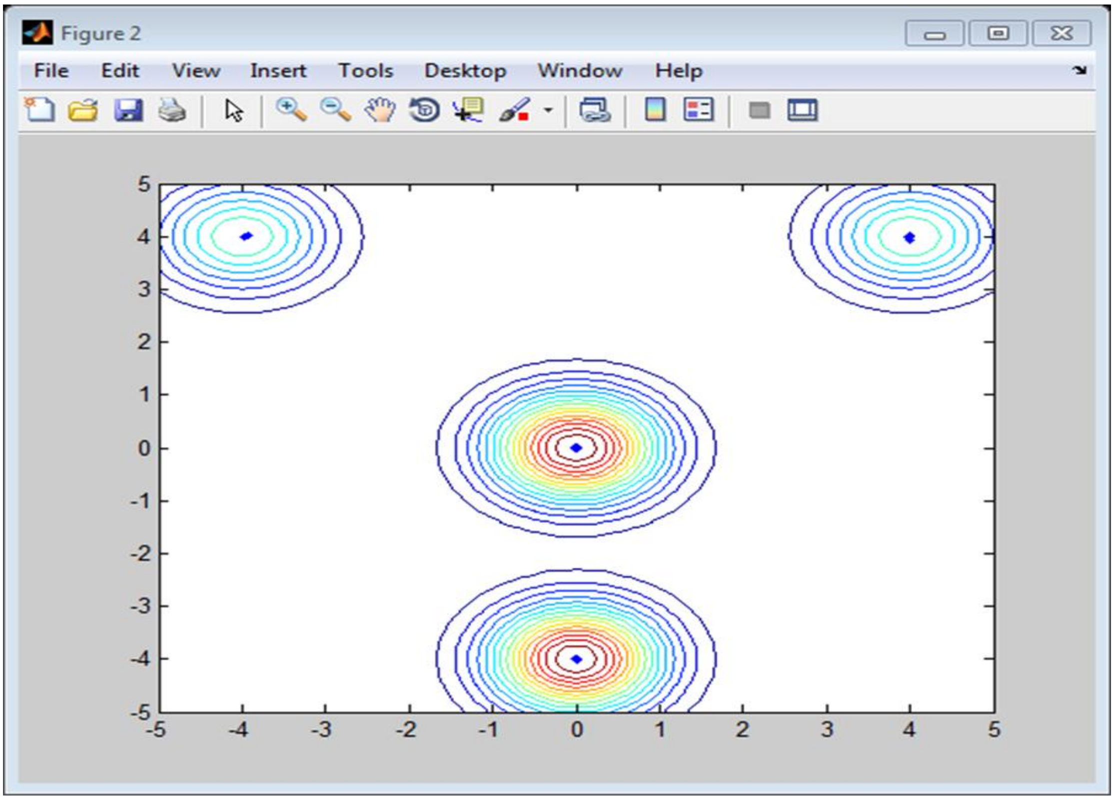

Fig: 4.18 Odor localization

As for case-2 with input number of flies and maximum iterations, all the odor sources are localized. Below figure will show the fitness value of FA and PSO during the time of odor localization. According to the number of flies and maximum iterations, fitness value has been calculated. The best fitness value will result when the flies is near to the odor source. With the given input values, calculating the fitness value and comparisons for the two algorithms has been evaluated. From the comparisons least required fitness value is calculated. This is the fitness value obtained in case-2 with input number of flies as 25 and maximum iteration times or termination condition as 80.

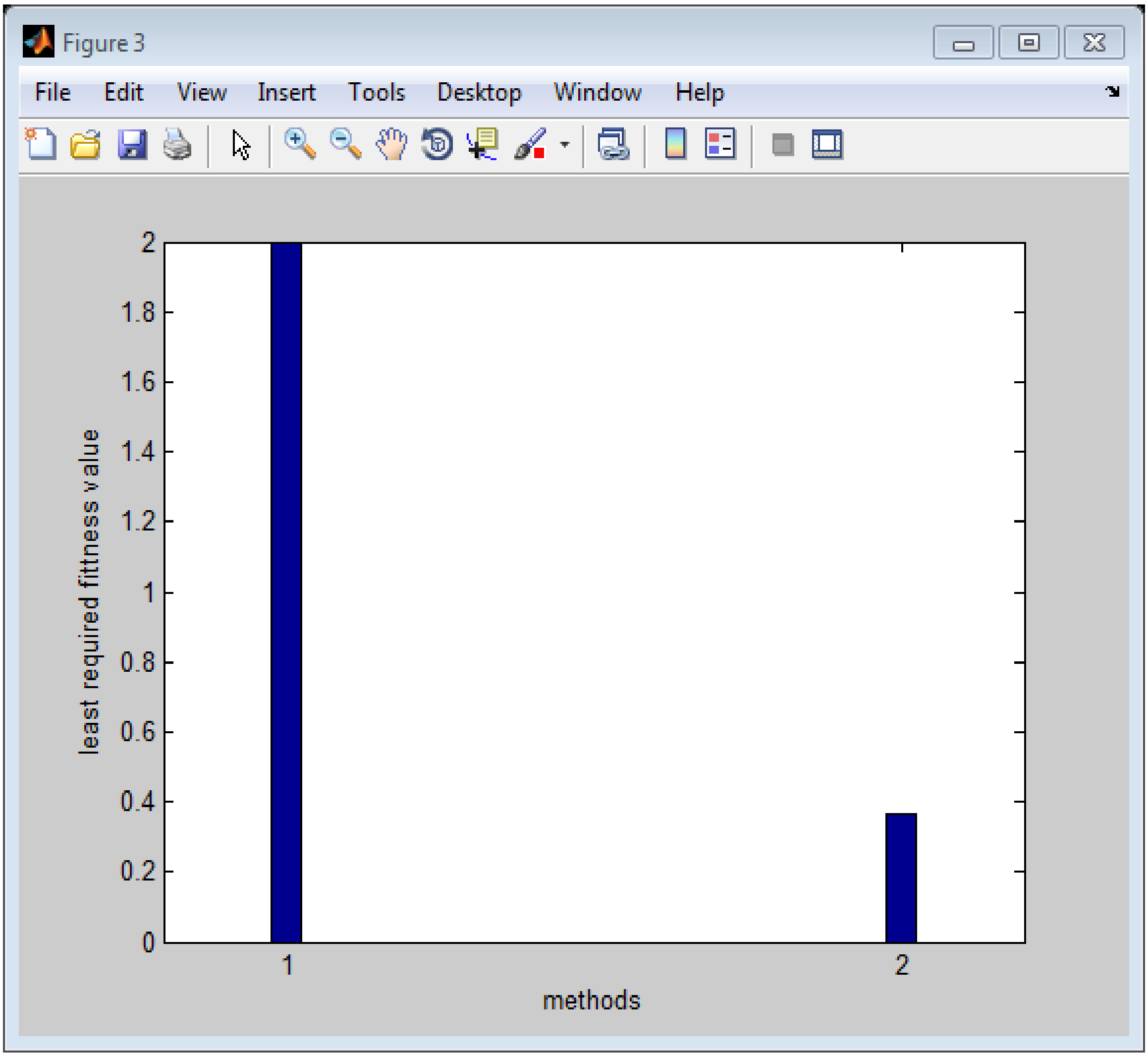

Fig: 4.19 Fitness value

Below shown figure will show the intensity of flies taken to localize odor sources

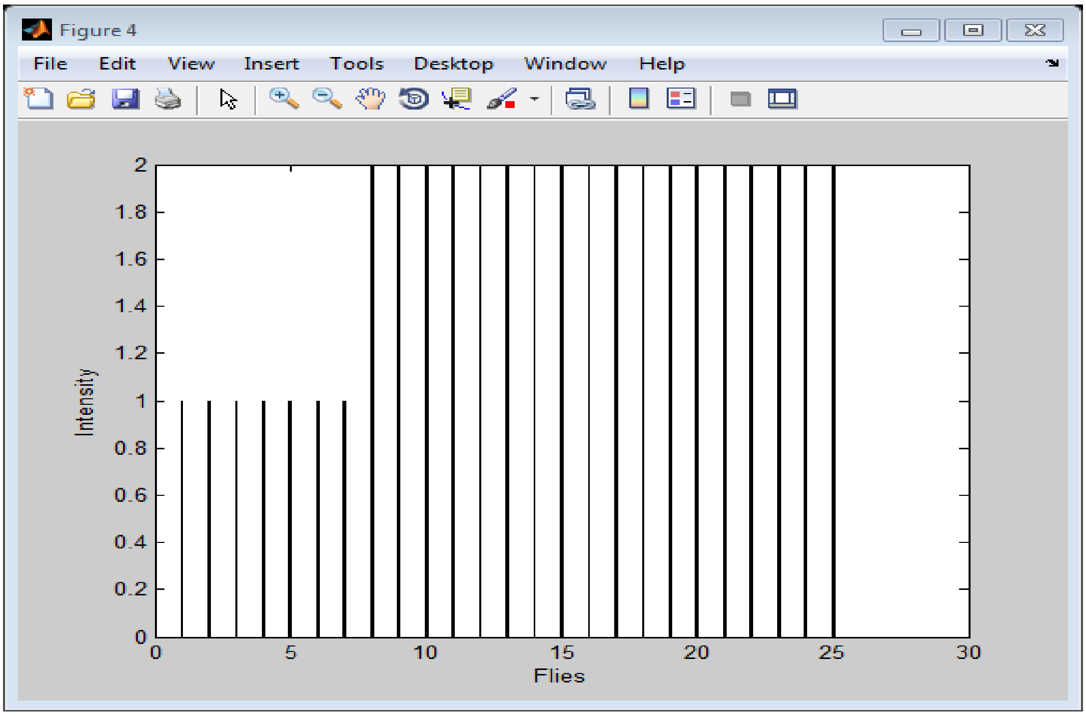

Fig: 4.20 Intensity of the flies

In the above figure, the first few flies has same range of light intensity, and as the iteration continues searching for the target, the flies intensity increases. The brighter ones will easily get attracted towards the odor source because brightness is associated with the objective function. Also, the less bright intensity will get attracted accordingly.

Below shown figure is showing the time consumption of FA with PSO

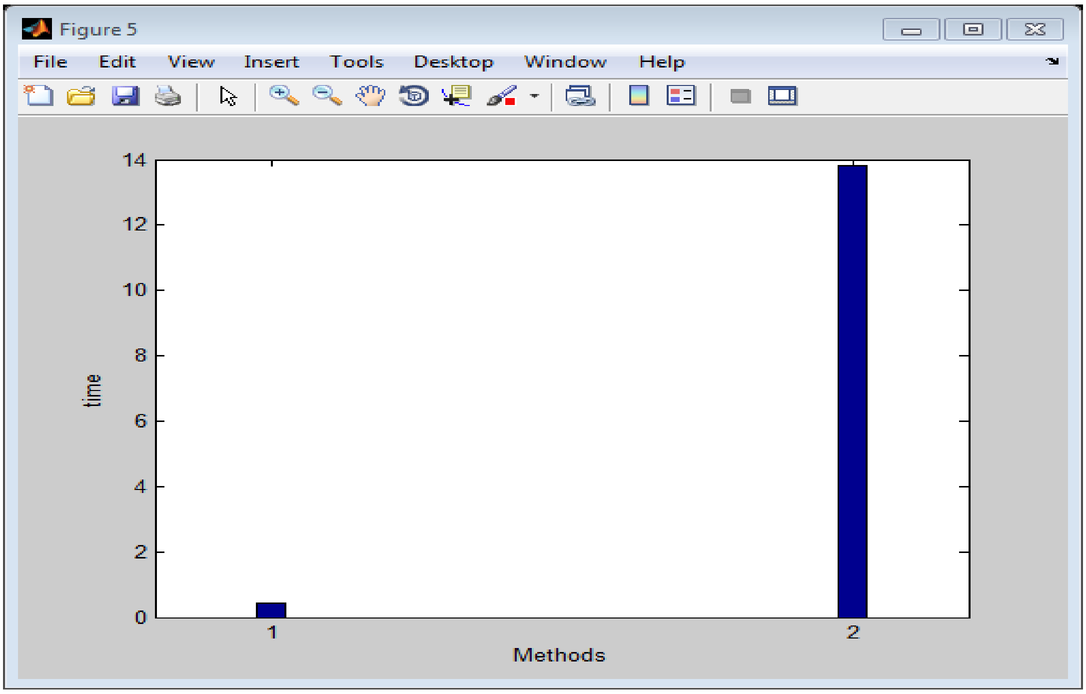

Fig: 4.21 Time Consumption

Figure below shows the collision rate comparison of proposed algorithm with the existing algorithm. Collision rate is calculated when multi-robot is localizing odor sources.

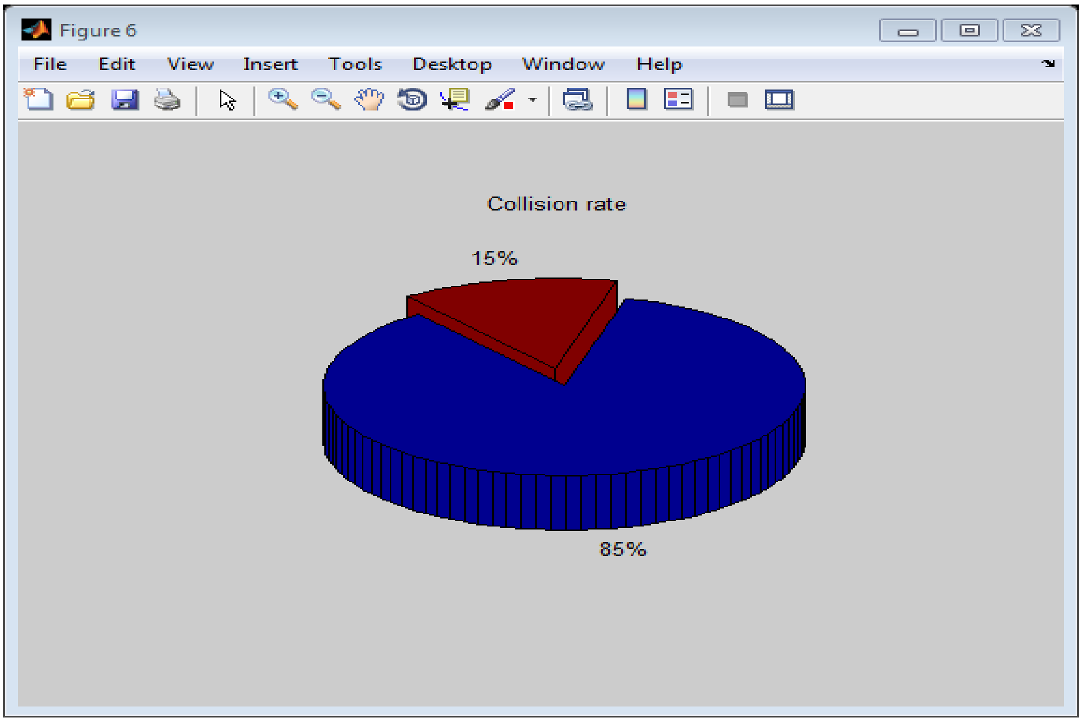

Fig: 4.22 Collision Rate

<u>Case: 3</u>

No of flies = 30

Maximum iteration = 100

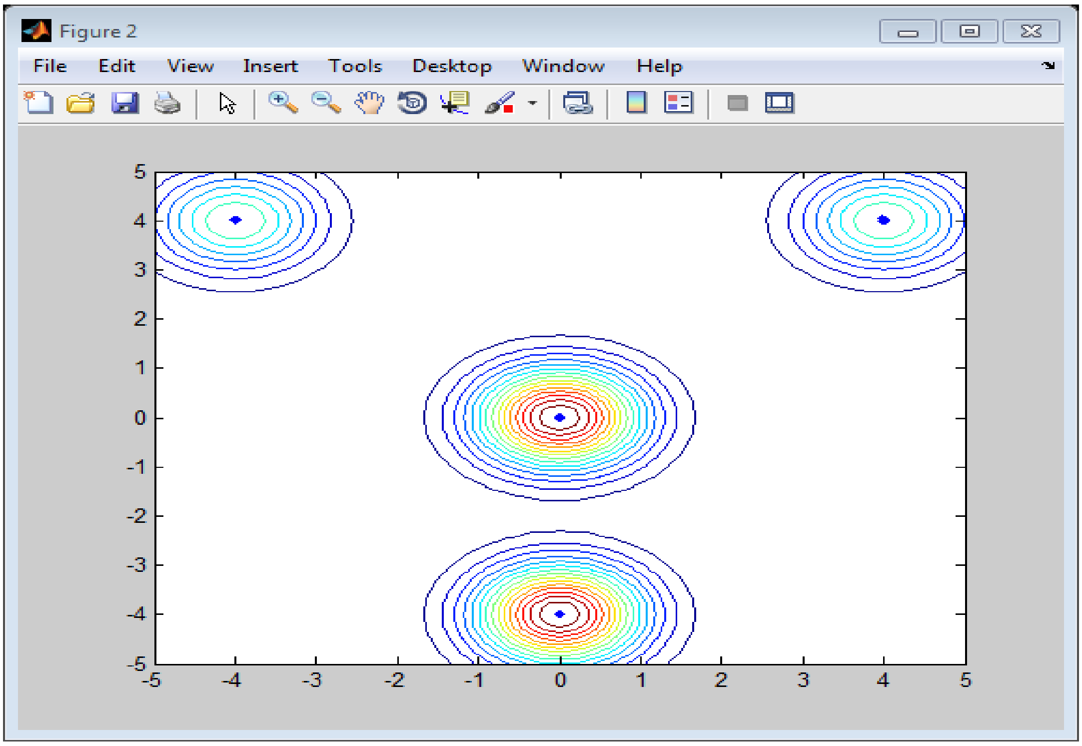

Fig: 4.23 Odor Localization

Figure below shows the comparative fitness value of FA and PSO during the odor localization process. Maximum iteration times will result in clear output and best fitness value. The best fitness values are obtained when the flies are near to the odor source. With the given input values, calculating the fitness value and comparisons for the two algorithms has been evaluated. From the comparisons, least required fitness value is calculated. Here, the fitness value of firefly algorithm is best than the particle swarm optimization because flies are near to the target i.e. the flies will localize the odor sources. As in case of particle swarm optimization, least fitness value is obtained because the swarms are not near to the target source.

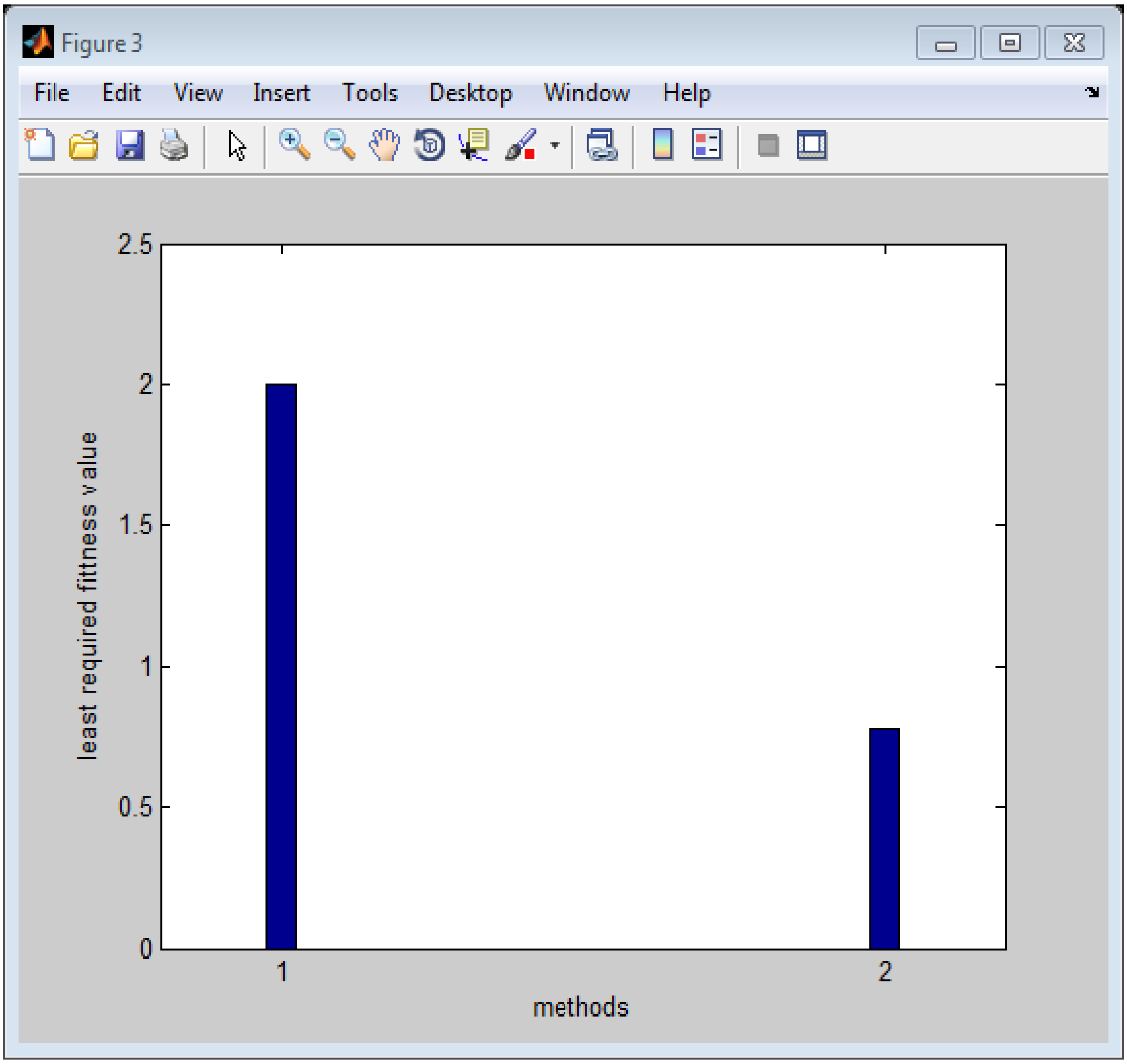

Fig: 4.24 Fitness value

Below figure shows the Intensity of the given flies for odor localization.

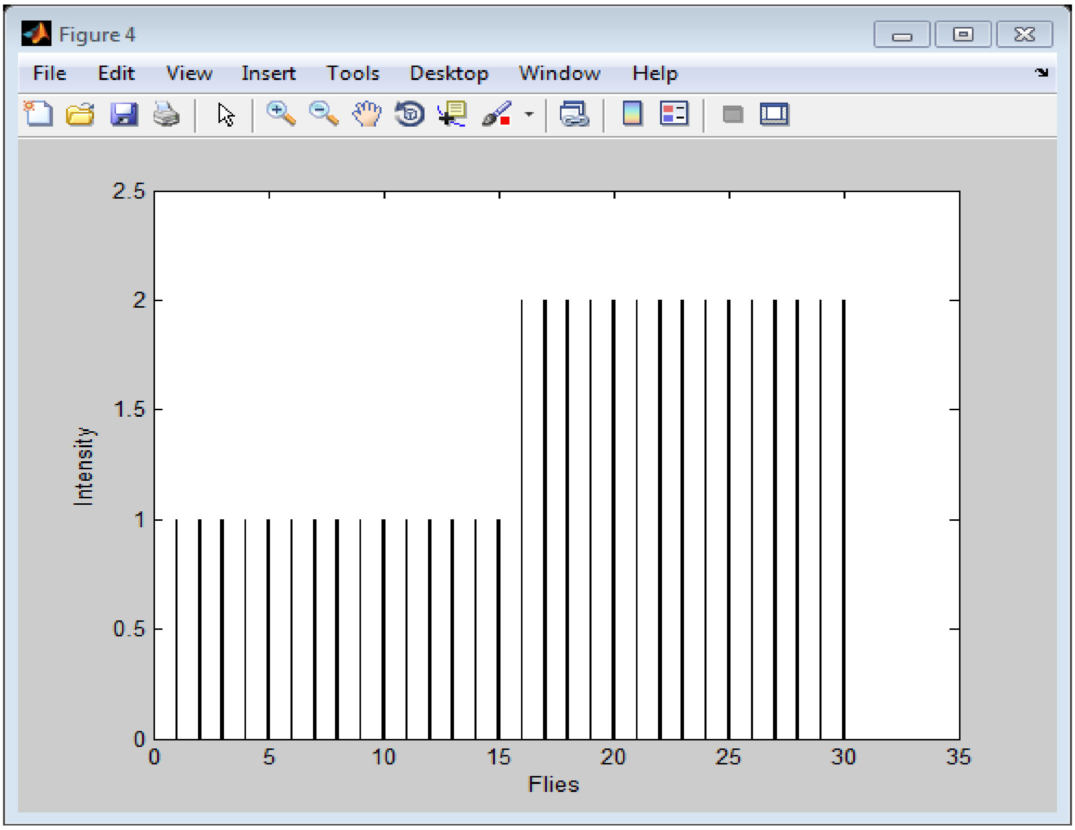

Fig: 4.25 Intensity of the given flies

Figure shows the time consumption taken when localizing odor sources using FA and PSO algorithm.

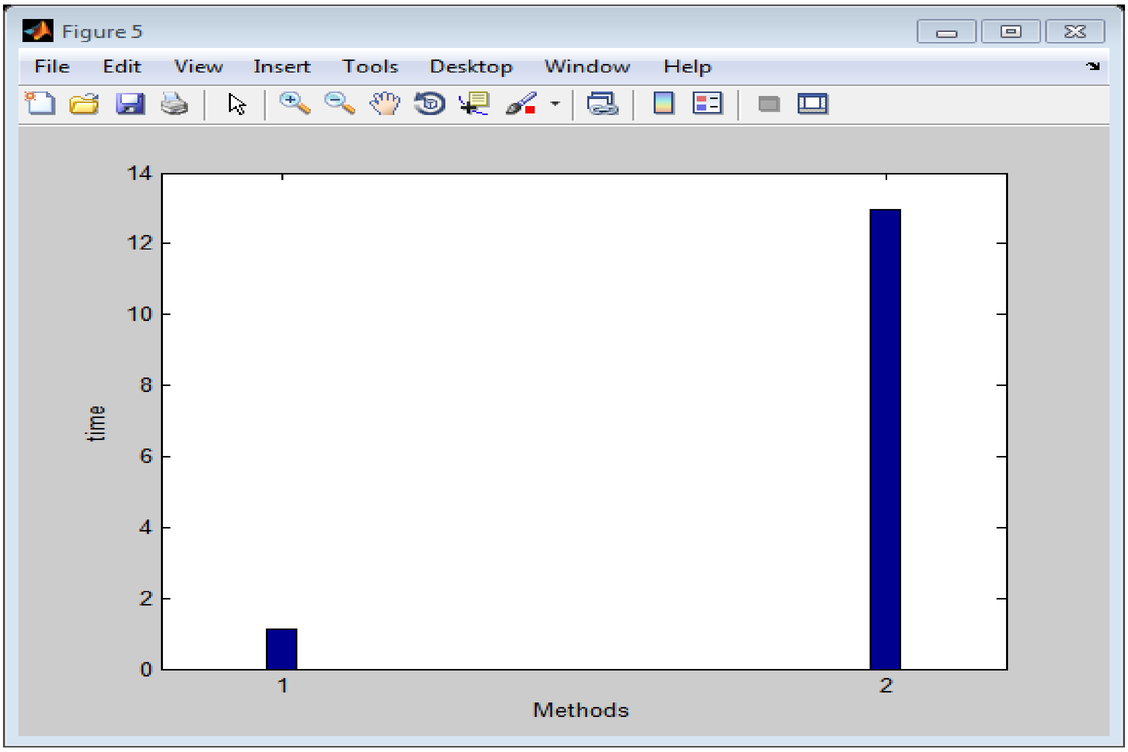

Fig: 4.26 Time Consumption

Figure below shows the collision rate comparison of proposed algorithm with the existing algorithm. Collision rate is calculated when multi-robot is localizing odor sources.

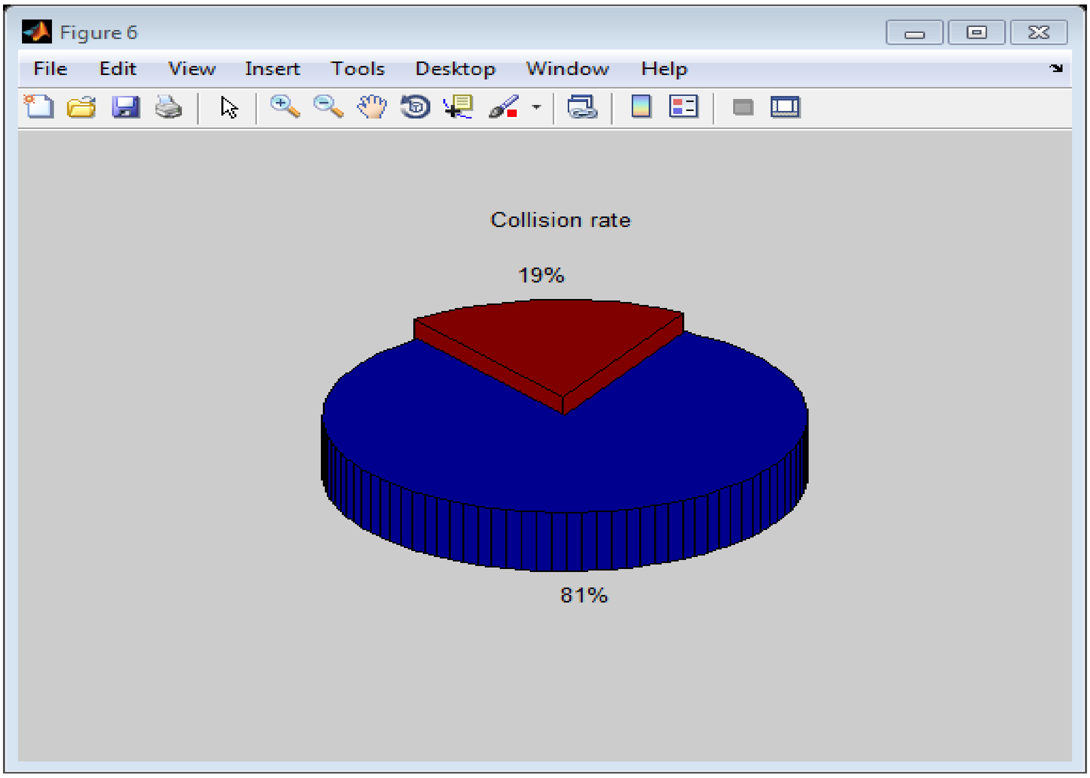

Fig: 4.27 Collision Rate

COMPARISON TABLE BETWEEN FA AND PSO

The comparative table between firefly algorithm and particle swarm optimization will measure the parameters of fitness value, time consumption in terms of multiple number of flies or swarm size and termination condition.

Below table will show the comparative analysis between these two algorithms in terms of some parameters and shows that the proposed algorithm is more efficient than the existing algorithm.

Table 1. Firefly Algorithm

No of flies	Termination condition	Fitness value (least)	Time
12	50	0.8	0.5
25	80	0.4	0.5
30	100	0.7	1.0

From the above table, calculating the average fitness value and time consumption of firefly algorithm

Average fitness value = 0.8 + 0.4 + 0.7/3

= 1.9/3

= 0.63

Average time consumption = 0.5 + 0.5 +1.0/3

= 2/3

= 0.67

Table 2. Particle Swarm Optimization

Swarm size	Termination condition	Fitness value (least)	Time
12	50	2	6.1
25	80	2	13.8
30	100	2	13

From the above table, calculating the average fitness value and time consumption of particle swarm optimization.

Average fitness value = 2 + 2 + 2 /3

= 6/3

= 2

Average time consumption = 6.1 + 13.8 +13/3

= 32.9/3

= 10.97

Above table shows the comparative analysis in terms of same number of flies or swarm size and maximum iteration times, to calculate some parameters and compare them using firefly algorithm and particle swarm optimization. With the experiment performed, the proposed algorithm is proved to have better performance than particle swarm optimization. Below figure shows the graphical presentation of our approach and existing work.

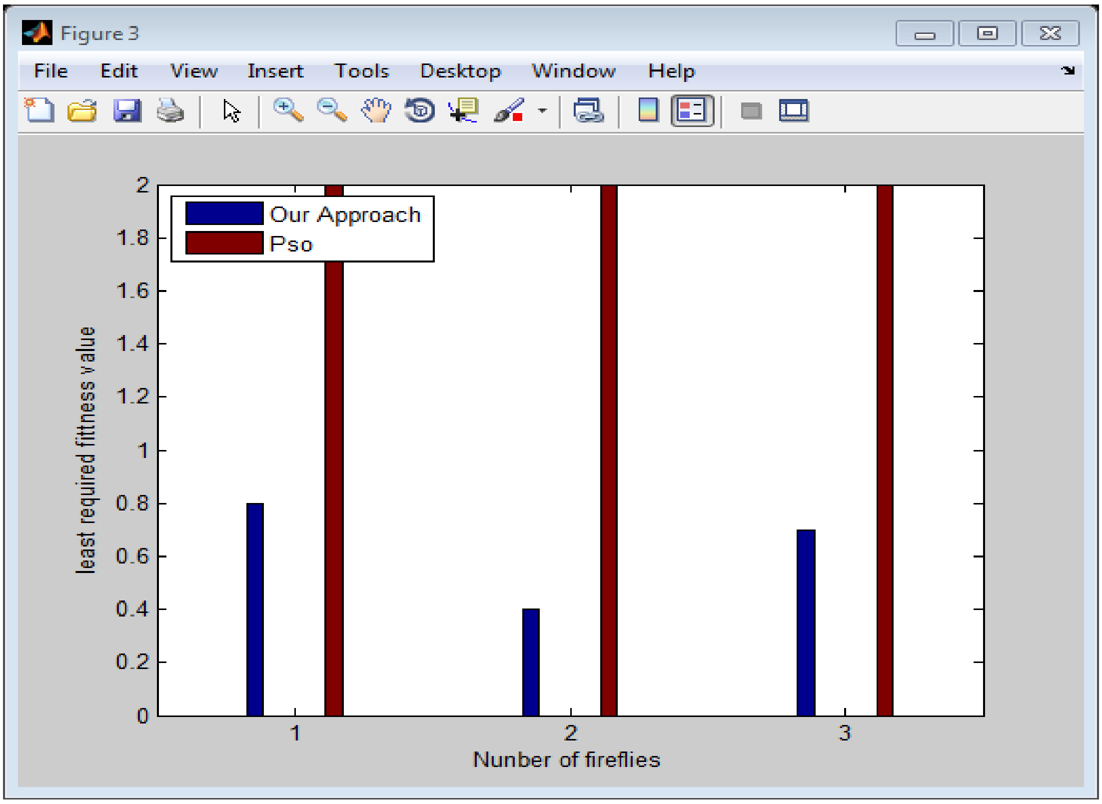

Fig: 4.28 Graphical presentation of our approach work with PSO

COLLISION PERFORMANCE

The comparison between collision rates taken in localizing odor sources with several input number of flies and maximum iteration times, comparing in terms of both the proposed firefly algorithm and particle swarm optimization. The collision rate is different for different inputs. In the movement phase of localizing odor sources, each robot must not collide with any other robot. By collision, a lot of time is wasted and that unable to achieve optimized results. Also, collision avoidance can be taken as a research work for further perspective.

The collision rate between the two algorithms shows that the collision rate of particle swarm optimization is much higher compared to the collision rate of the proposed algorithm for localizing odor sources by multi-robot cooperation method in a niche environment. Thus, performance of the proposed firefly algorithm is more efficient than particle swarm optimization.

CHAPTER 5
CONCLUSION AND FUTURE SCOPE

CONCLUSION

The problem is focus on firefly algorithm for localizing odor sources by multi-robot cooperation method based on a niching environment. A niche is formed, where particles in the niche start searching for odor sources. Compute light intensity and calculate the fitness value. As attractiveness is proportional to the brightness, firefly with less brightness will get attracted to a firefly with more brightness. With the number of iterations, robots will localized all odor sources, resulting in some parameters set up and also certain collision rates. Various experiments have been performed using *matlab*. Comparative measures are evaluated for certain parameters between proposed algorithm and PSO. Thus, experimental results show that the proposed firefly algorithm evaluates average fitness value and time consumption more efficient than particle swarm optimization for localizing all odor sources.

FUTURE SCOPE

This study of localizing odor sources can be further research in terms of localizing odor sources in a noisy environment, using more than one sensor in a robot and also Collision Avoidance as another research field. The proposed algorithm can also show an account for real-world environments, a further research field.

CHAPTER 6
LIST OF REFERENCES

[1] Afaq. H and Saini. S (2011) "Swarm Intelligence based on Soft Computing Techniques for the solutions to Multiobjective Optimization Problems", International Journal of Computer Sciences Issues (IJSCI), Vol. 8, Issue 3, No. 2, May (2011), pp. 498-510

[2] Blum. C, Merkle.D (Eds.) (2008) Swarm Intelligence Introduction and Applications, Springer-Verlag Berlin Heidelberg

[3] Borenstein. J and Koren. Y (1989) "Real-time Obstacle Avoidance for Fast Mobile Robots", IEEE Transactions on Systems, Man and Cybernetics, Vol. 19, No. 5, Sept./Oct. (1989), pp. 1179-1187

[4] Caro. D. G, Ducatelle. F and Gambardella. L. M (2005) "Swarm Intelligence for Routing in Mobile Ad-hoc Networks", In: Proc. of Swarm Intelligence Symposium, IEEE Publications, June (2005), pp. 76-83

[5] Chatterjee. A, Mahanti. G. K and Chatterjee. A (2012) "Design of a fully Digital Controlled Reconfigurable Switched Beam Concentric Ring Array Antenna using Firefly and Particle Swarm Optimization Algorithm" In: Proc. Electromagnetics Research B, Vol. 36, Aug. (2012), pp. 113-131

[6] Honarpisheh. Z and Faez. K (2013) "An Efficient Dorsal Hand Vein Recognition Based on Firefly Algorithm", International Journal of Electrical and Computer Engineering (IJECE), Vol. 3, No 1, Feb. (2013), pp. 30-41

[7] Hu. Q and Han. Q (2011) "Decision making in a multi-robot system for odor source localization" In: Proc. Of IECON 2011-37th Annual Conference on IEEE Industrial Electronics Society, Nov. (2011), IEEE publications, pp. 74-79 (2011)

[8] Ishida. H, Nakayama. G, Nakamoto. T and Moriizumi. T (2005) "Controlling a Gas/Odor Plume-Tracking Robot based on Transient Responses of Gas Sensors", IEEE Sensors Journal, Vol. 5, No. 3, June (2005), pp. 537-545

[9] Jatmiko. W, Pambuko. W, Mursanto. P, Muis. A, Kusumoputro. B, Sekiyama. K and Fukuda. T (2009) "Localizing Multiple Odor Sources in a Dynamic Environment using ranged sub-grouped PSO with flow of wind Based on Open Dynamic Engine LibrarySwarm", IEEE production (2009), pp. 602-607

[10] Kaur. G and Singh. R (2014) "Sharpening Enhancement of Ultrasound Images using Firefly Algorithm", International Journal of Advanced Research in Computer Science and Software Engineering, vol. 4, Issue 8, Aug. (2014), pp. 1039-1044

[11] Lochmatter. T and Martinoli. A (2009) "Theoretical Analysis of Three Bio-Inspired Plume Tracking Algorithms", IEEE International Conference on Robotics and Automation, May (2009), pp. 2661-2668

[12] Lu. Q, Liu. S and Qui. X (2010) "A Distributed Architecture with two layers for odor source localization in multi-robot systems", In: Proc. of Congress on Evolutionary Computation (CEC), 2010 IEEE publications, July (2010), pp. 1-7

[13] Marjovi. A, Nunes. J, Sousa. P, Faria. R, Marques. L (2010) "An Olfactory based Robot Swarm Navigation Method", 2010 IEEE International Conference on Robotics and Automation, May (2010), USA, pp. 4958-4963

[14] Nurmaini. S, Tutuko. B, Thoharsin. A (2013) "Intelligent mobile olfaction of swarm robots", International Journal of Robotics and Automation (IJRA), volume 2, No.4 Dec. (2013), pp. 189-198

[15] Yang, Sin-She (2010) Nature Inspired Meta-heuristics Algorithms Second Edition, Luniver Press

[16] Yang. X and He. X (2013) "Firefly Algorithm: Recent Advances and Applications", arxiv: 1308.3898 v1 [Math.OC], Aug. (2013), pp. 1-14

[17] Zhang. J, Gong. D and Zhang. Y (2013) “A niching PSO-based multi-robot cooperation method for localizing odor sources”, Elsevier (2013), neurocomputing 123, pp. 308-317 (2013)

[18] Zheng. Y, Liang. H, Xue. J, Chen. S, (2014) “Population Classification in Fire Evacuation: A Multiobjective Particle Swarm Optimization Approach”, IEEE Transactions on Evolutionary Computation, Vol. 18, No 1, Feb (2014), pp. 70-81

[19] http://www.tutorialspoint.com/matlab/matlab_useful_resouces.htm

CHAPTER 7
APPENDIX

LIST OF ABBREVIATIONS

SI: - Swarm Intelligence

PSO: - Particle Swarm Optimization

FA: - Firefly Algorithm

GSO: - Glowworm Swarm Optimization

ACO: - Ant Colony Optimization

RSSI: - Received Signal Strength Indicator

MPSO: - Modified Particle Swarm Optimization

ACO: - Ant Colony Optimization

ABC: - Artificial Bee Colony

CS: - Cuckoo Search

IWD: - Intelligent Water Drops

FSS: - Fish School Search

ODE: - Open Dynamics Engine