Discovering Heuristic Knowledge in an Ant Colony Optimization Technique

Mohd Razif Bin Shamsuddin, Nordin Abu Bakar, Rosnawati Abdul Kudus

Faculty of Computer and Mathematical Sciences
Universiti Teknologi MARA (UiTM)
40450 Shah Alam Selangor, MALAYSIA
razif@tmsk.uitm.edu.my, nordin@tmsk.uitm.edu.my

ABSTRACT

Heuristic knowledge is an instinct naturally embedded in animals for navigational purposes. Ant Colony Optimization (ACO) has captured this mechanism into a well defined technique to solve TSP and routing problems. Now, ACO gives a general framework to solve other problems in similar nature. This paper tends to explain how knowledge discovery is done in ACO and tested the modified ACO in solving robot's path planning problem. The results have been encouraging for ACO to produce a good navigational path for a robot to follow but the performance has been inferior to fuzzy approach used in the same problem domain. The results are discussed in detail and the comparative findings are justified clearly focusing on the unique criteria of the chosen ACO technique.

Keywords
Heuristics, Ant Colony, Optimization, Fuzzy Logic, Path Planning Problem

1.0 THE ACO MODEL

Ants are sometime annoying especially when they invaded our kitchen. They will go all over the place to search for food. But before you dig under your kitchen cabinet for the ant killer, consider how a group of ants can teach us to solve problems. Try a simple experiment by breaking the trail made by those ants from behind the wall to your sugar jar. Make a line with your finger so that the trail is broken. You will see in an instance that those ants at the back of the trail will disperse and loose direction. But before you know it the broken trail will be amended (maybe slightly off track) and they continue the work as usual. This is the very ability that has stunned scientists – the ability to find the shortest path.

1.1 Knowledge discovery by means of updating pheromone trails

The pheromone trails are updated after all the ants have constructed their tours. The strategy is used to provide string additional reinforcement to the arc belonging to the best tour found since the start of the algorithm. Thus, pheromone evaporation is implemented by an equation (1.1a) (Dorigo & Stutzle, 2004), where 0<ρ<1 is the pheromone evaporation rate. The parameter ρ is used to avoid unlimited accumulation of the pheromone trails and its enable the algorithm to explore new trails. After evaporation, all ants deposit pheromone on the arcs they have crossed in their tour as given in equation (1.1b) where $\Delta\tau_{ij}^{k}$ is defined as in equation (1.1c) and $\Delta\tau_{ij}^{bs}$is defined as in equation (1.1d) (Dorigo & Stutzle, 2004). In equation (1.1c), C^k is the length of the tour, T^k, built by the k^{th} ant which is computed as the sum of the length of the arcs belonging to T^k. Thus, the better ant's tour has more pheromone on then tour arcs and therefore the arcs are more likely to be chosen by ants in the future iterations of the algorithm. In equation (2.1d), the elitist strategy provides strong additional reinforcement to the best tour so far by adding a quantity e/C^{bs} to its arcs where e is a parameter that defines the weight given to the best so far by tour T^{bs} and C^{bs} is its length (Dorigo & Stutzle 2004)..

$$\tau_{ij} = (1-\rho)\cdot\tau_{ij} \qquad (1.1a)$$

$$\tau_{ij} = \tau_{ij} + \sum_{k=1}^{m}\Delta\tau_{ij}^{k} + e\Delta\tau_{ij}^{bs} \qquad (1.1b)$$

$$\Delta\tau_{ij}^{k} = \begin{cases} 1/C^{k} & ,if\ arc(i,j) belongs\ to\ T^{k} \\ 0, otherwise \end{cases} \qquad (1.1c)$$

$$\Delta\tau_{ij}^{bs} = \begin{cases} 1/C^{bs} & ,if\ arc(i,j) belongs\ to\ T^{bs} \\ 0, otherwise \end{cases} \qquad (1.1d)$$

There are three ideas from natural ant behaviour that are simulated in the ACO model :

1)The preference for paths with a high pheromone level[4] .
2)The higher rate of growth of the amount of pheromone on shorter paths.
3)The trail mediated communication among ants.

[4] Pheromone level is the secretion produced by ants when they make a trail.

2.0 ACO IN ROBOT PATH PLANNING PROBLEM

The use of ACO in TSP gives a general framework to solve other problems in similar nature. The same framework will be used to solve robot's path planning problem. The aim of the problem is find the shortest tour while avoiding the obstacles (Gil et al, 1990; Firby, 1987). Robot landscape is defined as a two dimensional grid with 100 by 100 squares . This is a simulated workspace that will be used throughout this experiment to implement ACO as an engine to drive the robot from a defined starting position to a final target destination. In between there will be obstacles to test the vulnerability of the ACO algorithm in maneuvering the robot to avoid collision (Kaufman, 1987).

(x-1,y+1)		*(x+1,y+1)*
	(x,y)	
(x-1,y-1)		*(x+1,y+1)*

Figure 2.1: An ant current position (bold) with 4 possible next positions (italic)

An ACO is considered a global path planning strategy where it works by having a complete knowledge about the environment. It searches paths within the valid region and establish a connection between a start state with a goal state. A global planner stops the search when a valid path is found or no path is detected (Majdi & Soleimanpour, 2008).

2.1 Knowledge Robot Navigation and Obstacle Avoidance Strategy in ASO

Robot navigation knowledge algorithm must be implemented in order to allow successful obstacle avoidance. In the design process a robot navigation strategy is designed as illustrated in Figure 2.1.

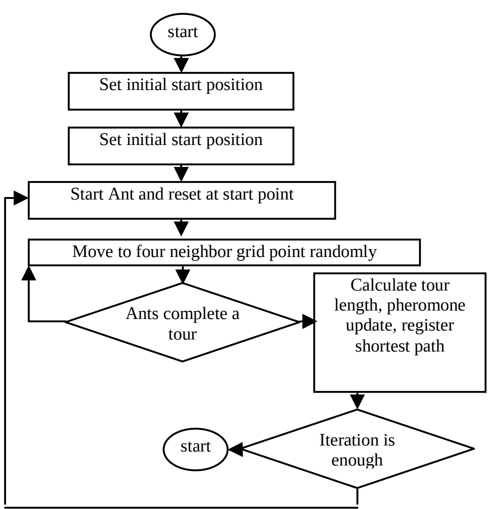

Figure 2.1:Flowchart for robot navigation and obstacle avoidance strategy

3.0 THE EXPERIMENTS

Figures should be labeled with "Figure" and tables with "Robot path planning problem is one of the interesting problems that have been the aim of many researches in artificial intelligence. In this study, we focus on how the robot can be manipulated to learn the surrounding landscape. For these experiments, there are three primary landscapes that have been the testing ground; The Big Hall, The Wall Following and the Volcano Challenge.

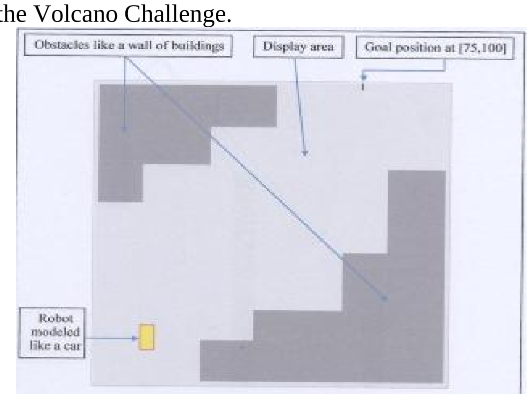

Figure 3.1: The Big Hall Set Up

The big Hall set up is supposed to be the simplest task that a robot could be assigned for. A goal-seeking behavior is adopted to find the target. This task resulted maximum performance for both learning algorithms.

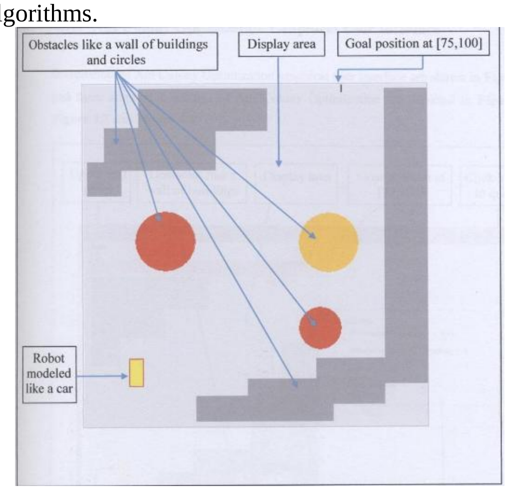

Figure 3.2: The Wall Following Set Up

The Wall Following set up is aimed at the added challenge for machine learning capability. The simulated robot will need to recognize he obstacles and try to avoid it while keeping the target in the search. The robot was stationed at different starting positions prior to the search and find activities. These different starting points are reflected in the results that will be shown in the later part of this report. Basically the farther the robot being stationed has a direct impact on the performance for one algorithm but insignificant to another one.

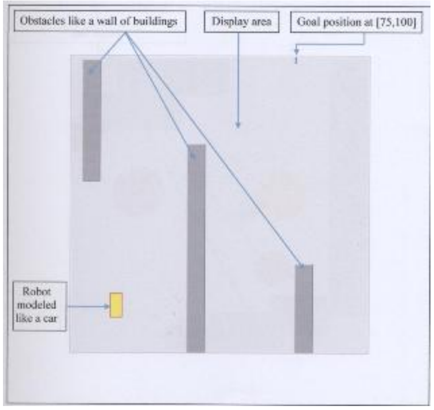

Figure 3.3: The Volcano Challenge Set Up

The volcano challenge gives some added obstacles to increase the difficulty level for machine learning algorithms to drive the robot to the target area.

4.0 THE RESULTS

The performance for the two learning algorithms ; ant colony optimization (ACO) and Fuzzy Logic Approach (Fuzzy), are measured with the distance taken to reach destination and time performance to find the target.

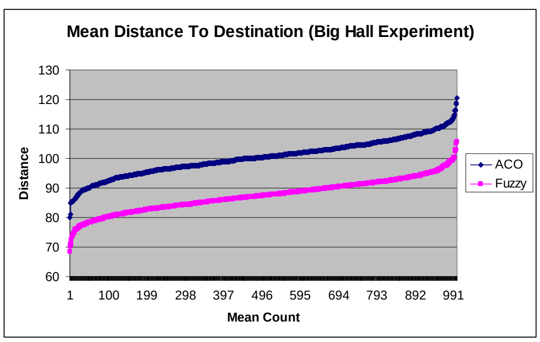

Figure 4.1: The Big Hall Experiment

The results of the Big Hall experiment is shown in Figure and it shows how the Fuzzy approach required lesser distance to reach the destination. The means were produced from 20 starting positions.

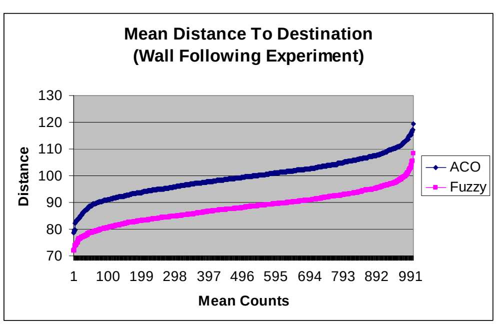

Figure 4.2: The Wall Following Experiment

The results for the wall following experiment also gives an indication that the fuzzy approach performed better by requiring lesser distance to reach the goal.

The volcano challenge shows the results that the fuzzy approach is better than the ACO learning algorithms.

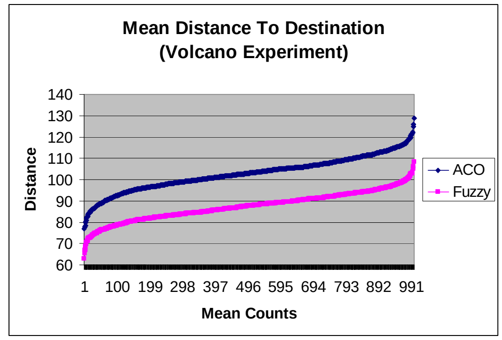

Figure 4.3: The Volcano Challenge

4.1 Findings

The comparison results of the proposed methods for distance traverses from the initial position to the goal position as shown in Figures 4.1,4.2,4.3, reveal that both proposed methods performances are satisfactory for goal seeking, wall following and obstacles avoidance behaviours. Besides, the proposed methods are able to find the shortest path from initial position to the goal position while avoiding obstacle constellations in a reasonable time.

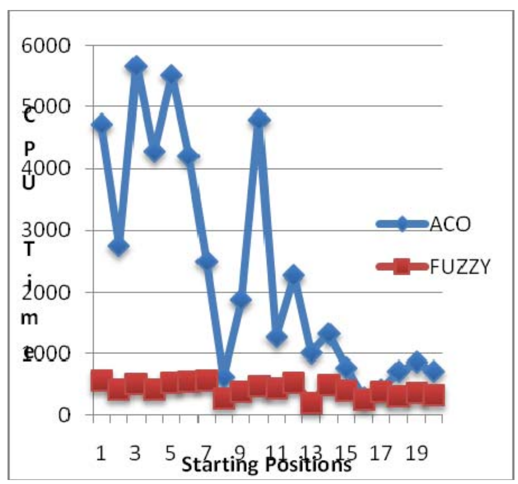

Figure 5.1: Time performance for big hall experiment

The results of the experiments using the proposed methods for CPU time consumptions to complete the task are shown in Figures 5.1, 5.2 and 5.3. It is shown that in most cases, the Fuzzy approach has consumed less CPU computations time compared to ACO. In addition, the overall performance of Ant Colony Optimization on three simulations models as mentioned before shows that Ant Colony Optimization consumed CPU time double than the Fuzzy approach. A reasonable explanation is that Ant

Colony Optimization algorithm works by reinforcing good solutions., therefore, more CPU time is consumed in order to generate good solutions.

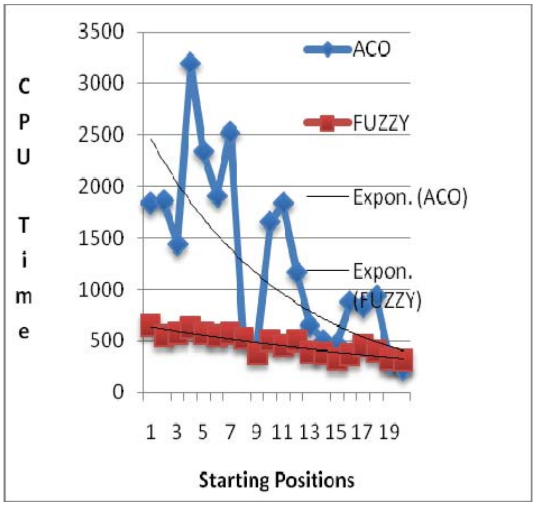

Figure 5.2: Time Performance for the wall following experiment

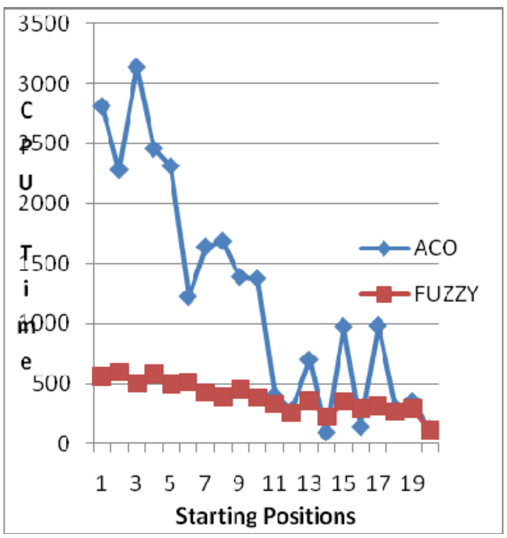

Figure 5.3: Time performance for the volcano challenge.

ACO in many cases. The Fuzzy approach performances are better in terms of distance traverses from the initial position to the goal position CPU time consumptions to complete the task for goal seeking, wall following and obstacles avoidance behaviours. This is due to the fact that the fuzzy based navigation strategy employed sensor to guide local planner navigation and minimizes collision with the stationary obstacles.

In addition, the Fuzzy approach adopted navigation method which is based on a confined sensor region surrounding actual state and do not consider the entire state space. As the result, the amount of computation for Fuzzy Controller System which is employed by the Fuzzy approach is reduced by using only the nearest obstacles to determine the robot direction.

5.0 DISCUSSION

This paper has discussed in detail the experimental results of the Fuzzy approach and Ant Colony Optimization for robot path planning problem. The performances of the proposed methods were evaluated on goal seeking, wall following and obstacles avoidance behaviours which was measured by distance traverses from the initial position to the goal position and CPU time consumptions to complete the task on three simulation models . Each of the simulation models has been designed with different environment and stationary obstacles constellations.

Ant Colony Optimization which employed Elitist Ant System does not performance as good as Fuzzy Controller System on many problem instances. Such a result is due to the fact that the Ant Colony Optimization adopted global navigation strategy. The global path planning works by having complete knowledge about the environment. It searches a path inside the region of valid configurations, thereby connecting a start state with a goal state. Therefore, the exploration of the robot's workspace considers the entire state space that leads to very time consuming process.

REFERENCES

Abu Bakar, Nordin. (2001). AI Technologies in Business. *Konferensi Akademik, UiTM Caw Terengganu, Kuala Terengganu.*

Althoefer K. (1996). *Neuro-Fuzzy Path Planning for Robotics Manipulatos.* PhD Thesis. King's College London, U.K,

Anton, H. (1995). Calculus with Analytic Geometry. *John Wiley & Son.*

Brooks, R. A. (1986). A robust Layered Control System for a Mobile Robot. *IEEE Journal of Robotics and Automation, 2*(1), 14 – 23.

Dorigo,M., & Stutzle,T. (2004). The Ant Colony Optimization. *MIT Pres.*

Dorigo,M. et.al. (1996). IRIDIA. *Universite Libre de Bruxelles, Belgium, IEEE.*

Erdmann, M., & Lozano-Perez. (1987). On Multiple Moving Obstacles. *Algorithmica 2*(4), 477 -521.

Firby, J. R. (1987). An Investigation into Reactive Planning in Complex Domains. *AAAI Conference.*

Fugimura, Kikuo. (1991). Motion Planning in Dynamic Environments. *Springer Verlag, Tokyo, Japan*.

Gat, E. (1991). *Reliable Goal-Directed Reactive Control for Real-World Autonomous Mobile Robots*. PhD Dissertation. Virginia Polytechnic Institute and state University.

Gil de Lamadrid, J., & Gini, Maria. (1990). Path Tracking through Uncharted Moving Obstacles. *IEEE Transactions on System, Man, and Cybernetics*, *20*(6), 1408 – 1422.

Goldberg, D.E. Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley, Reading, Mass. (1989).

Griswold, N. C., Elan, J., " Control of Mobile Robots in the Presence of Moving Obstacles ", IEEE Transactions, on Robotics and Automation, 6(2), 1990.

Holland, J.H. Adaptation in Natural and Artificial Systems, University of Michigan Press, Ann Arbor, MI (1975).

Kaufman, Morgen, " An Architecture for Intelligent Reactive Systems " , 1987.

Khatib, O., " Real-Time Obstacle Avoidance for Manipulators and Mobile Robots " , The International Journal of Robotics Research, 5(1), pp. 90, 1986.

Kosko, B., " Neural Networks and Fuzzy System: A Dynamical Systems Approach to Machine Learning ", Addison Wesley, 1992.

Majdi,M., Hassan,S. A. And Soleimanpour, S.Multi AGV Path Planning in Unknown Environment Using Fuzzy Inference Systems, IEEE,2008.

Negnevitsky, Micheal, " Artificial Intelligence: a guide to intelligent systems", Addison Wesley, Page 97, 2005.

Ratering, S., & Gini. (1995). Robot Navigation in a Known Environment with Unknown Obstacles. *Autonomous Robots*, 149-165.

Sugeno, M., & Nishida, M. (1985). Fuzzy Control of a Model Car. *Fuzzy Sets and Systems, 16*, 103 - 113,

Topalov, A. V., & Tzafestas, S. G. (2000). Layered Multi-Agent Reactive Behaviour Learning in a Robotic Soccer. *Proc. SYROCO.2000,Vienna, Austria.*

Munakata, T., Jani, Y. (1994). Fuzzy System: An Overview. *Communication of the ACM, 37*(2), 69-75.

Tzafestas, S. G. (1999). *Advances in Intelligent Autonomous Systems.* Boston: Kluwer.

Yen, J., & Pflunger, N. (1992). A Fuzzy Logic Based Robot Navigation System. *AAAI Fall Symposium on Mobile Robot Navigation, Boston*, 195.