Robot global path planning overview and a variation of ant colony system algorithm

Buniyamin N., Sariff N., Wan Ngah W.A.J., Mohamad Z.

Abstract**—**This paper presents the results of a research that aims to develop an algorithm to solve robot path planning (RPP) problems in static environments. The problem is to find a global optimal path that satisfies the optimization criteria of shortest path with minimum computation time. A description of a variation of Ant Colony System (ACS) algorithm utilized for Robot Path Planning (RPP) purposes is presented. A representation of heuristic and visibility equation of state transition rules is proposed to sustain the function of Ant Colony System (ACS) for solving RPP problem of finding the optimal path. This algorithm was applied within a global static map that consists of feasible free space nodes.

The performance of the algorithm in terms of computation time and number of iteration required to obtain an optimal path were evaluated by using a simulation approach. Subsequently, its performance was compared to the performance of Genetic Algorithm (GA) a well known and established RPP algorithm. The results obtained indicate that the developed algorithm performed much better than the GA. In addition, an overview of robot path planning (PP) algorithms in global static environment is also offered.

Keywords**—**Path Planning, Ant Colony System, Algorithm, Global Environment, Autonomous Robot.

I. INTRODUCTION

THE ability to avoid obstacles is an important design requirement for any mobile robots. Thus, the determination of a collision free path between start and goal positions through obstacles cluttered in a workspace is central to the design of an autonomous robot path planning [1, 2]. Path planning can be defined as “the determination of a path that a robot must take in order to pass over each point in an environment and path is a plan of geometric locus of the points in a given space where the robot has to pass through” [3-6]. Path planning research covers a wide area of robotics research because it enhances robotic navigation systems in both static and dynamic environments. With the perfect path planning system, mobile robots can navigate by itself without human intervention to reach the targeted destination.

Robot navigation problems as shown in Figure 1 can be generalized into four categories which are 1) localization, 2) path planning and 3) motion control and 4) cognitive mapping. Among these problems, it can be argued that path planning is the most important issue in the navigation process. Path planning enables the selection and identification of a suitable path for the robot to traverse in the workspace area. The two main components for global or deliberative path planning are 1) robot representation of the world in configuration space (C-space) and 2) the algorithm implementation. These two components are interrelated and greatly influence one another in the process to determine an optimal path for the robot to traverse in the workspace within an optimal time [2].

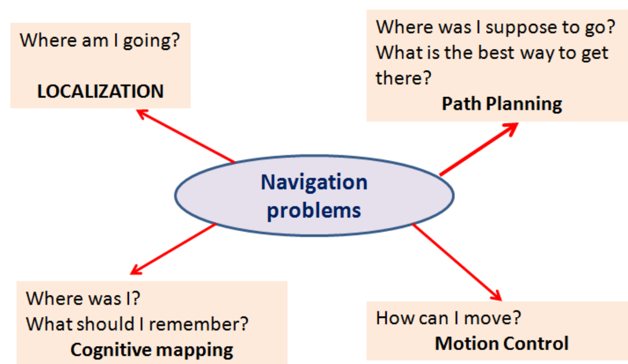

Fig. 1 Robot navigation problems.

If the knowledge of the environment is known, the global path can be planned offline before the robot start to moves. This global path can aid the robot to traverse within the real environment because the feasible optimal path has been constructed within the environment. However, another category of path planning system known as local path planning was introduced to solve RPP problem when the robot is faced with obstacles. Local path is usually constructed online when the robot avoids the obstacles in a real time environment [1, 3]. Figure 2 shows the different between the two mobile robot navigation approaches.

Currently many autonomous robotic systems have integrated path planning algorithms in one system. The combination of the main two approaches and multiple path planning algorithms in one system is called hybrid path planning method. Hachour [7] for example, used multiple path planning algorithms in one system. In the development of his Hybrid Intelligent Systems (HIS) for Intelligent Autonomous Vehicles IAV in unknown environment, he successfully combined Genetic Algorithms (GA), Fuzzy Logic (FL), Neural Networks (NN) and Expert Systems (ES).

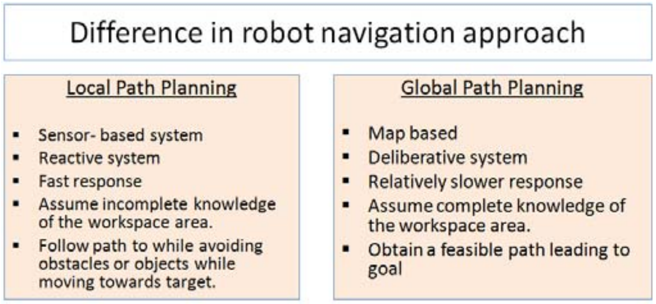

Fig.2 Mobile robot path planning approaches.

Due to the significant of the RPP system, many researchers have focus on the RPP problem in both static and dynamic environments. This research is focused on RPP problem in global static map with the goal to find the optimal path and also satisfy the optimization criteria. The algorithm proposed in this research is develop based on the analogy of behavior of ants while finding its food known as the variation of Ant Colony System (ACS) created for RPP purposes.

ACS is within the swarm technology category. Swarm technology or systems mimic's natural systems where many individuals coordinate and work together to achieve a target by using decentralized and self-distributed organization approach. Swarm technology is thus based on social habits and work ethics of insects such as ants, bees and certain types of fish and birds. These creatures' works as a team to achieve their target. Based on the behavior and ability of colonies of ants that are able to find the shortest route between their nest and a food source, Marco Dorigo [8] developed an optimization algorithm known as ACS or Ant Colony Optimization (ACO) algorithm in early 1990's. The algorithm proposed herewith is a variation of the ACO algorithm.

II. Path Planning Algorithms

The evolution of robot global path planning algorithms from 1980 until today shows that there are numerous type of path planning algorithms proposed by researchers to solve RPP problem [2, 9]. The numerous research in this area lead to an improvement of the global path planning approaches from one generation to the next generation. It started with finding the path to goal successfully. Then the global RPP system is enhanced by finding the path successfully and must, at the same time satisfy a certain optimization criteria where for example, the computation time and path cost of the algorithms while finding its path must be considered.

A. Global Path Planning

In global PP, both the complete description of goal and static space of the obstacles are made available. The objective is to find a collision free path for the robot to move from initial position to goal position. By using the created global path, before it moves, the robot will have a reference and guidance to which way it can traverse the course to goal point or position.

Figure 3 depicts a framework to represent Global PP. As shown, a model of the map of robot workspace area including the location of robot, obstacles and free space area known as configuration space (C-space), must first be created. In C-space, all possible configurations of robot are represented. Then, the C-space will be modeled by discretizing the free space area to construct a graph that represents the connectivity of the space. This graph is constructed by using an appropriate algorithm that presents the connectivity of the graph which also known as graph search techniques. Finally, the PP algorithm is applied within this graph to find a feasible path for the robot to reach the goal.

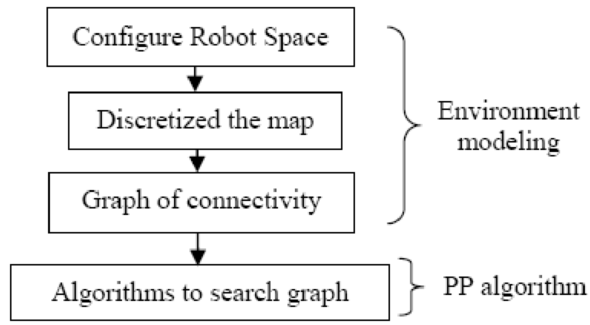

Fig. 3 Framework of graph search techniques for robot global PP

B. Evolution of Global Path Planning

Path Planning Algorithms (PPAs) have progressed and its evolution shows many improvements from one generation to the next generation. The more recent generation of PPAs have been created to be more adaptive and able to work within the robot environment itself [10].

The development and evolution of traditional path planning approaches such as Artificial Potential Field [11], Neural Network [12], Distance Wave Transform [13], A* algorithm [14], D* algorithm [9] and etc, proposed by previous researchers have changed and evolved to other variation of path planning approaches that is based on approaches categorized as artificial intelligence [15].

These approaches are also known as evolutionary computation. An example, is the Genetic Algorithm (GA) [16], which is an algorithm created from the analogy of behavior of ants such as Ant Colony Optimization algorithm [8] and algorithm from behavior of swarm intelligences such as Particle Swarm Optimization algorithm [8, 17]. These algorithms are not only capable to find optimal paths that satisfies the optimization criteria but is also adaptable and robust in both static environments and also in dynamic environments [18-21]. Compared to traditional approaches, these methods have been proven as a robust and effective search technique that can be used to optimize the RPP problem [18].

Since its appearance in approximately 1992 [8], ACO has been used in solving many optimization problems such as the Traveling Salesman Person (TSP) problem [22]. ACO is a search technique inspired by the foraging behavior of real ants. With ACO ability to solve a hard combinatorial optimizations problem, the use of ACO has contributed to the success of many RPP research. For example, Tan Guan Zheng [23] proposed the use of ACS to find robot path based on map of MAKLINK graph. Results of their study show that ACS performances are better than evolutionary approaches. Hao Mei et al [24] then combined ACO with Artificial Potential Field to produce the path planning in dynamic environment. Gengqian et al [19] have verified that ACO can find an optimal path in their grid map by proposing its own state transition rules equation. A literature study shows that the application of ACO to solve the RPP problems has not been explored in detail.

The research presented herewith has been carried out to examine the performances of ACS in a given global map. The ACS is proposed with a combination of a new state transition rules to solve the RPP efficiently with support from the actual local and global updating rules of ACS. The performances of the algorithm in terms of computational efficiency was observed and evaluated based on the distance, time and number of iteration the algorithm takes to find an optimal path compared to GA. The goal is to enhance knowledge of Ant Colony Optimization algorithms in RPP research area. In this paper, the mapping and path planning algorithms construction including the pseudo code is first discussed. Then, the results and discussions are provided. Finally, a conclusion that compares and summarizes the performances of ACO is presented.

III. Research Methodology

In the initial stage, the environment was mapped using an appropriate mapping algorithm that represents the global workspace area where the robot works. This map will represent the start point, goal point, and differentiate the area which consists of obstacles and free space before the path planning algorithm can be applied. In this research, the map complete with connection of free space nodes was constructed by assuming that the obstacles have been ignored at the early stage of the path planning process. The nodes are located at x-y coordinates and this was the input nodes the algorithm will work with. With this available map, the path planning process has been simplified where it only need to optimize the feasible free space path to goal.

In order to plan the path, the ACS will use these nodes as an input for it to find path to goal efficiently. Representation of nodes with integer numbers will be used during the initialization process while the real location of x-y coordinate will be used to evaluate exactly the path cost or distance between one node to another node. Finally, the optimal path and the performances evaluation will be recorded to verify the effectiveness of the proposed algorithm itself. Fig. 4 illustrates the proposed method applied within this experimental research study.

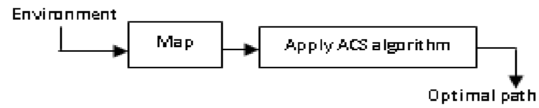

Fig. 4 Proposed Method

A. Environment Modeling

For this research, a 2D global map was created with nodes that are of a 10 X 10 cm size. A map that consists of 26 free space nodes connected to one another has been proposed. This map was created by assuming that obstacles have been ignored at the early stage of map construction as shown in Fig. 5. As a result, the map will only consists of feasible free space nodes as illustrated in Fig. 6. Similar maps have been proposed by previous researchers such as the Visibility graph [13], random graph [21] and MAKLINK graph [23].

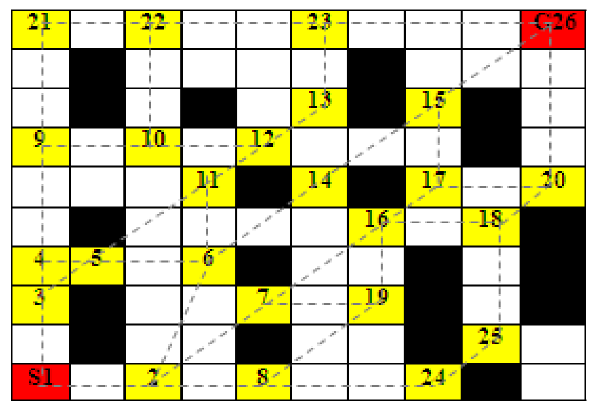

Fig. 5 Example of global feasible map

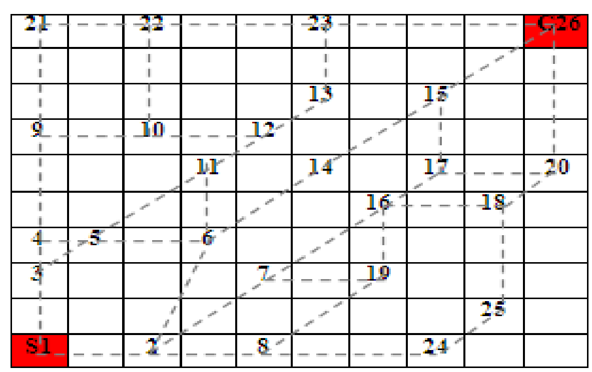

Fig. 6 Global free space map used by ACS

During the construction of this map, the location of free space nodes (yellow cell) was determined randomly without applying any specific mapping algorithm. The location of free space feasible map was determined randomly by referring to the knowledge provided on the map itself. The area where the robot can safely traverse including it size is represented by a white cell while the boundary area of obstacles includes the safety region is represented by black cells as shown in Figure 3. At last, using the final model of the map depicted in Fig. 4, the algorithm will start to find a solution by initializing the population of feasible path (blue line) or unfeasible path (red line) to goal as shown in Figure 7, during the construction process of finding an optimal path. Figure 8 and Figure 9 are examples that show the population of a path.

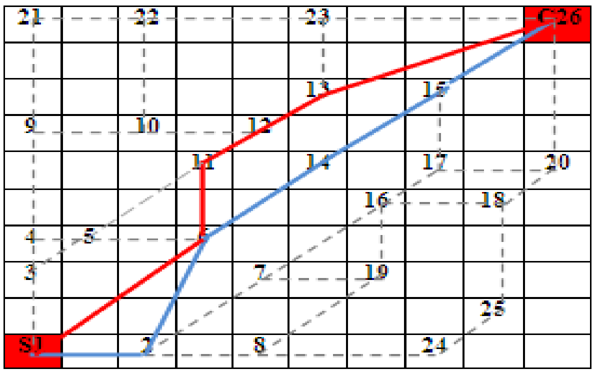

Fig. 7 Feasible and unfeasible path

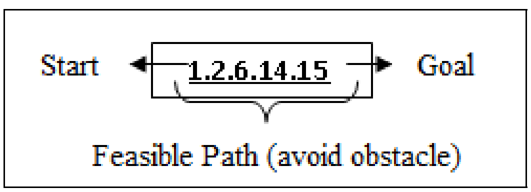

Fig. 8 A sample of feasible path population

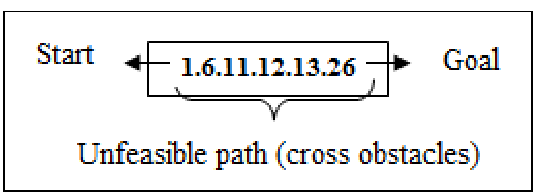

Fig. 9 A sample of unfeasible path population

IV. Ant Colony System Algorithm

For this RPP purpose, the proposed path planning algorithm is a modification of the original ACO concept (also known as Ant Colony System) proposed by Marco Dorigo [11]. Figure 10 outlines the implementation of ACO for RPP of a mobile robot. The model and concept of the proposed algorithm is as follows:

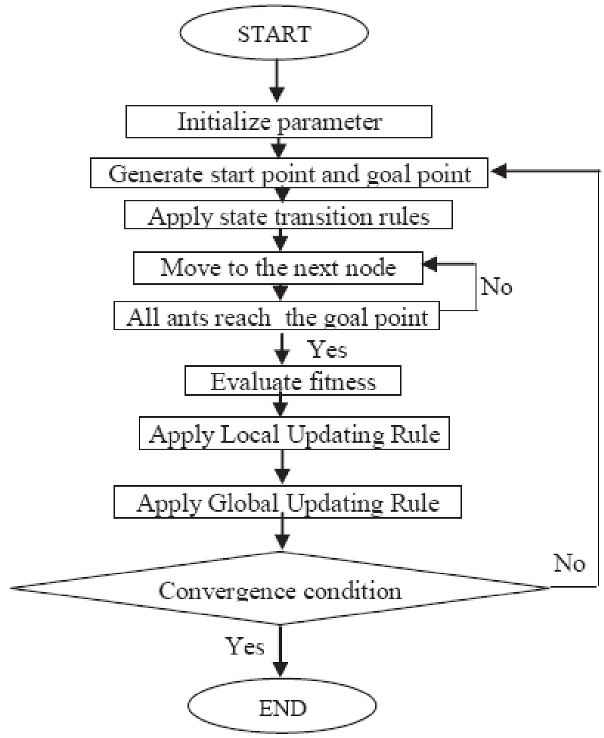

Fig. 10 Outline of ACO for RPP of a mobile robot

Step 1: Starting from the start node located at x-y coordinate of (1,1), the ant will start to move from one node to other feasible adjacent nodes provided during the map construction as shown in Fig. 11. As depicted, there are three possible movements available for the ants to move from the start node (1,1). The possible nodes are: node 1 located at the x-y coordinate of (1,3), node 2 at (2,3) and node 3 at (3,2).

Step 2: The ant will then take the next step to move randomly based on the probability given by equation (1) below:

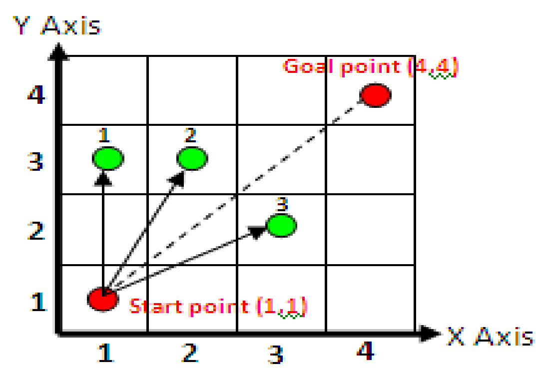

Fig. 11 Ant current position (red nodes) with 3 possible next positions (green nodes)

Probabilityij (t) = Heuristic ij (t)*Pheromone ij (t)
=[(1/distance between vector start point to next point and start point to reference line to goal)β* (trail/Σ trail)σ]
(1)

where Heuristicij (t) indicates every possible adjacent nodes to be traversed by the ant in its grid position at every t time. The quantity of Pheromone ij (t) is an accumulated pheromone the between nodes when the ant traverse at every t time. Therefore, the probability equation depends on both values where it will guide the ants to choose every possible node in every t time.

A. Derivation of heuristic@ visibility equation

The heuristic equation was derived from the idea of the local and global search solution for the RPP purposes. This represents the distance between selected adjacent nodes with intersect point at references line where the intersect point must be perpendicular 900 to the next node (Distance A) as shown in Fig.12. Due to the objectives of this research, the distance between each node that have the shortest distance should have a higher probability to be chosen compared to the longer distance. To ensure that this is achieved, the derivation of heuristic proposed must be inversed with the distance found as shown in equation (2) below:

Heuristic=[1/distance between next point with intersect point at reference line]β (2)
*where β=heuristic coefficient

Fig. 12 depicts the proposed solution of using the Pythagoras Theorem to calculate the distance A. For this purpose, the outline of the triangle has been developed to simplify the derivation. Using Pythagoras Theorem, Sin θ is equal to Distance A/Distance B where Distance A is equal to Sin θ* Distance B. θ here refers to the angle of vector of intersect point to next visited node with line of intersect point to goal position while Distance B refers to the distance from intersect point at references line to the next visited node. Intersect point is the point obtained when the line of the next node which is parallel with the x-axis intersects with the references line. Thus, the equation for distance A is equal to:

Distance A=Sin θ*Distance B=Sin θ* sqrt [(y2-y1) 2+(x2-x1) 2] (3)

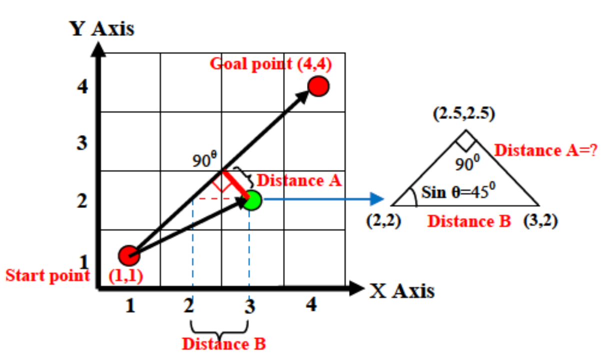

Fig. 12 Derivation of Distance A

As depicted in Fig. 12, the heuristic calculation can be calculated as shown below:

Distance A= Sin θ* sqrt [(y2-y1) 2+(x2-x1) 2]
= Sin 450* sqrt[(2-2) 2+(3-2) 2]
= 0.7071*1= 0.7071

Visibility=[1/0.7071]1=1.4142

B. Derivation of trail equation

The amount of pheromone or trail will also determine the probability of the next node to be chosen. The artificial ants behave similar with the real ants where the pheromone will attract the next ants to follow the shortest path until the process converges when all ants follow the same path to target. Equation (4) is the trail equation used in this research.

Trail= [trail//Σ trail] α ,where α=trail coefficient (4)

The concept of the amount of pheromone used is similar with the original ACS concept where the higher the amount of pheromone, the higher the probability value which will attract more ants to follow the short path and simultaneously cause the ACS to converge faster. Thus, if the amount of pheromone is smaller, the probability will become lower thus the path cost will be higher as the ants will avoid traversing a path with a lower pheromone amount.

Step 3: Each time ants construct a path from one node to another, the pheromone amount will be reduced locally by the given evaporation rate using the formula of update local rules as shown below:

Tij (new trail) ←(1-ρ)* τij (old trail), (5)
* where ρ=evaporation rate

This equation shows that each time the ants move from one node to another node, the amount of local pheromone will be updated in parallel. This process is important to prevent the map from getting unlimited accumulation of pheromone and enables the algorithm to forget a bad decision that has been previously made.

Step 4: Once the ants found its path to goal, the fitness of ants will be calculated. This covers the calculation of distance or path cost each ant takes to traverse from start point to goal point by using derivation of objective function for RPP below:

Distance= sqrt [(y2-y1) 2+(x2-x1) 2] (6)

Step 5: The fitness value will then be used for the process of global update. When all ants reach the destination, the ants will update the value of pheromone globally based on the fitness found by each ant by using Equation (7) below. This process will be repeated as the path being traverse by ants in each generation is determined using this global value. Generally, this process will attract the ants to follow the optimal path. During the process, the path with the shorter distance will be chosen as the probability to be chosen is higher compared to the path with the longer distance. The equation of global updating is derived in (7) and (8) below:

$$\tau_{ij} \leftarrow \tau_{ij} + \Sigma \Delta \tau_{ij}^{k} \quad (7)$$

where $\Delta \tau_{ij}^{k}$ is the amount of pheromone of ant m deposits on the path it has visited. It's defined as below:

(8)

where Q is number of nodes and Ck is the length of the path Pk built by the ants.

Step 6: The process will be repeated from Step 1 to Step 5 until the process converges. The process will stop when all ants traverse the same path that shows the shortest path to goal has been found. The detailed flow of the algorithm structure is simplified as shown in pseudo code below.

C. Pseudo Code of ACS Algorithm for RPP

```
If iteration (tmax )=1,2,3,4,5,6,…………………+n
    Else if ant(m)=1,2,3,4,5,6,…………………….+n
      Else if nodes(n)=1,2,3,4,………………….+n
     Compute the probability of the m th ants next nodes
    Move to the next nodes by computed probability
   Store history of past location of nodes in an array
      If current location of nodes is equal to destination
      Break the nodes (n) loop
     End
End
Evaluate fitness and store path distance of m th ants
Compute pheromone amount generated by m th ants
End
Update pheromone amount of the entire map
End
```

D. Implementation

The pseudo code was then translated and coded into MATLAB source code via a function available within MATLAB 7.0.4. The simulation was carried out using a computer with Intel (R) Celeron (R) M processor 1.5 GHz with 504MB of RAM. Various simulation results were then recorded based on the evaluation criteria required for this experiment outcomes such as optimal path, path cost, time, number of iterations, and etc.

V. Results and Discussions

A. ACS Performance for RPP

The result of implementing the ACS on an environment that consists of simple number of obstacles (shown in Fig.4) is tabulated in Table 2 below. The indications from the test results are that the ACO can find the optimal path successfully using selected parameter settings as shown in Table 1 below. The optimal path found by the ACS in 10 test runs is the same path with linkage of nodes 1.2.6.14.15.26 and the path cost is approximately 13.6476 cm as shown in Fig. 13 below. The average computation time is equal to 15.4062 sec while the average number of iterations is equal to 3.5 times.

The way ACS work is based on the behavior of ants where the heuristic and amount of pheromone plays an important role to guide the ants to reach the destination successfully. With the proposed state transition rules derived from the idea of creating the references line from start to goal point, ACS have been proven able to efficiently find an optimal global path. Similar to the original ACS concept, the path which has the higher amount of pheromone will attract more ants to traverse within that path while the path which has the low amount of pheromone will influences the behavior of ants to work avoiding the obstacles. Typically the short distance will cause the accumulated amount of pheromone to become higher compared to a longer distance path. The results in Table 2, indicates that the proposed ACS algorithm seems to be able to find an optimal global path.

Table 1: ACO Parameter Specifications

ACO Properties	*Properties*
Type of ACO	Ant Colony System
Population of ants	20
Maximum length of path	8
Pheromone coefficient, β	5
Heuristic coefficient, α	5
Evaporation rate, ρ	0.5
Convergence condition	Max-min of 20 pop≤0.0001
Maximum Iteration	40

Table 2: Computation Times and Iterations of ACO

No of run	*Optimal path*	*Distance*	*Time(sec)*	*Iteration*
1	1.2.6.14.15.26	13.6476	13.3536	3
2	1.2.6.14.15.26	13.6476	18.7286	4
3	1.2.6.14.15.26	13.6476	10.0510	3
4	1.2.6.14.15.26	13.6476	8.4564	2
5	1.2.6.14.15.26	13.6476	23.6816	5
6	1.2.6.14.15.26	13.6476	18.5721	4
7	1.2.6.14.15.26	13.6476	8.9377	2
8	1.2.6.14.15.26	13.6476	18.4917	4
9	1.2.6.14.15.26	13.6476	22.0616	5
10	1.2.6.14.15.26	13.6476	11.7273	3
Total Average			**15.4062**	**3.5**

C. Comparative Study of ACO and GA performance for RPP

Based on the results tabulated in Table 3, the optimal path found by both algorithms is the same as path found in Fig. 13 but the computation time and number of iteration is difference. ACO found the same path faster with a small number of generations in 5 test runs. The average time is 63 seconds and required only an average of 4.4 numbers of iterations. In the case of Genetic Algorithm (GA), the test run times have an average of 157.18 seconds with an average 8 number of iterations. This indicates that the proposed ACO is more robust and more effective. This is due to the guidance of the state transition rules that makes the ants work more intelligence compared to GA that is based on evolutionary approaches.

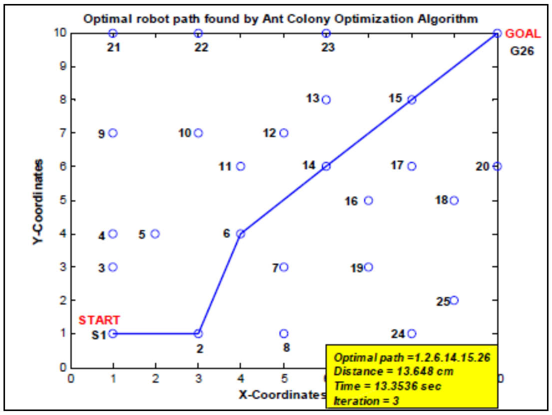

Fig.13 Optimal path found by ACS at the 3rd generation

Table 3: Computation Times for GA & ACO

RPP algorithms		*GA*		*ACO*	
No of run	*Optimal path & path cost*	*Time*	*Iteration*	*Time*	*Iteration*
1	1.2.6.14.15.26 (13.6476 cm)	111.838	10	104.606	4
2		147.958	7	44.4	4
3		114.362	8	73.552	6
4		310.464	7	43.635	4
5		101.278	8	49.297	4
Total Average		**157.18**	**8**	**63.098**	**4.4**

For the proposed ACS, the ants will determine the next chosen node accurately with the guidance of state transition rules where it can help the ants to traverse the nodes near the optimal nodes and ignore unfeasible nodes. In contrast, with GA, the next node to be chosen is based on random approaches that will simultaneously cause GA to go through the selection and remove process iterately while finding the optimal path. Thus as shown, ACO can find the optimal path faster within a smaller number of generations compared to GA. Figure 14 depicts the time each algorithm took to find the optimal path for each run and Figure 15 shows the number of iterations required to obtain the optimal path.

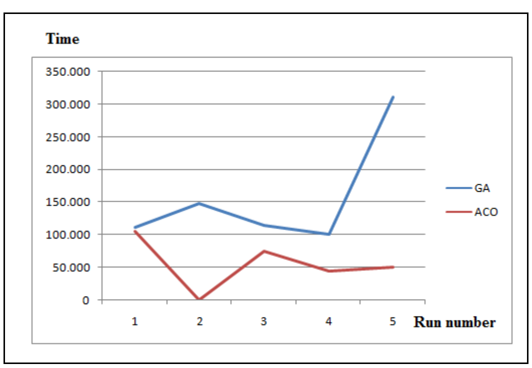

Fig.14 Time (seconds) required for algorithm to find optimal path.

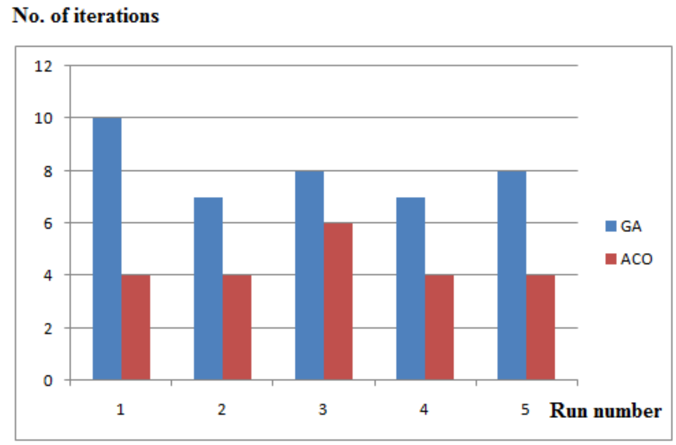

Fig.15 Number of iterations required to find optimal path.

VI. Conclusion

The proposed variation of ACS for RPP cases within a given map of feasible nodes was proven able to find the optimal path effectively. The optimal path found satisfied the optimization criteria for RPP purposes which are to reduce the path cost and shorter computation time with smaller number of generations. The comparative study between Genetic Algorithm and ACO also proved that ACO is more robust and effective in finding optimal path compared to GA. The research indicates that, the applications of ACS can be further explored to expand the applications of both optimization algorithms in RPP research area.

Acknowledgment

The authors gratefully acknowledge and thank University Technology MARA, Malaysia (UiTM) for providing a grant for this research. [600-RMI/ST/DANA 5/3/Dst (147/2009)].

References

[1] Susnea, I., Viorel Minzu, Grigore Vasiliu. Simple, real-time obstacle avoidance algorithm for mobile robots. in 8th WSEAS International Conference on Computational intelligence, man-machine systems and cybernetics (CIMMACS'09) 2009.

[2] Sariff, N., Buniyamin N. An Overview of Autonomous Mobile Robot Path Planning Algorithms. in Proceedings of 4th Student Conference on Research and Development (SCORED 2006), 27th -28th June. 2006. Shah Alam, Malaysia.

[3] J. Wan Ngah, B.N., Mohamad Z. Point to Point Sensor Based Path Planning Algorithm for Mobile Robots. in 9th WSEAS International Conference on System Science and Simulation in Engineering (ICOSSSE'10). 2010. Iwate, Japan.

[4] Sariff, N., Buniyamin N. Ant Colony System For Robot Path Planning In Global Static Environment. in 9th WSEAS International Conference on System Science and Simulation in Engineering (ICOSSSE'10),. 2010. 4th -6th October, Iwate, Japan

[5] Haklıdır, M., Taşdelen Modelıng And Sımulatıon Of An Anthropomorphıc Robot Arm By Usıng Dymola. in 5th Int. Symp. on Intelligent Manufacturing Systems. 2006.

[6] Vacariu, L., Flaviu Roman, Mihai Timar,Tudor Stanciu, Radu Banabic, Octavian Cret. Mobile Robot Path-planning Implementation in Software and Hardware in 6th WSEAS International Conference on Signal Processing, Robotics and Automation. 2007. Corfu Island, Greece.

[7] Hachour, O., The proposed hybrid intelligent system for path planning of Intelligent Autonomous Systems. International Journal Of Mathematics And Computers In Simulation, 2009. 3(3): p. 133-145.
[8] Dorigo, M., Stutzle T, Ant Colony Optimization: The Bradford Book 2004, London, England. : MIT Press Cambridge. Masachusetts,.
[9] Stentz, A. Optimal and Efficient Path Planning for Partially-Known Environments. in IEEE International Conference on Robotics and Automation. 1994.
[10] Sariff, N., Buniyamin N. Comparative Study of Genetic Algorithm and Ant Colony Optimization Algorithm Performances for Robot Path Planning in Global Static Environments of Different Complexities. in 8th IEEE International Symposium on Computational Intelligence in Robotics and Automation (CIRA2009) , December 15-18. 2009. Daejon, Korea.
[11] Khatib, O., Real-Time Obstacle Avoidance for Manipulators and Mobile Robots, 5, 1986. The International Journal of Robotics Research: p. 90-98.
[12] Xin, D., Hua-Hua C, Wei K.G., , Neural Network and Genetic Algorithm Based Global Path Planning in A Static Environment. Journal of Zhejiang University Science, 2005. 6A: p. 549-554.
[13] Zelinsky, A., Yuta S. A unified approach to planning, sensing and navigation for mobile robots. in 3rd Int. Symposium on Experimental Robotics. 1993. Kyoto, Japan.
[14] 14. Warren, C.W., Fast path planning using modified A* Method. IEEE Transactions on Systems, Man and Cybernetics, , 1993.
[15] Netnevitsky, M., Artificial Intelligence: A guide to intelligent Systems. 2nd Ed ed. 2002: Addison-Wesley.
[16] Goldberg, D.E., Genetic and evolutionary algorithms come on age. Communications ACM, 1994. 37: p. 113-119.
[17] Eberhart, Y., Shi, Particle swarm optimization: developments, applications and resources. in Evolutionary Computation. Proceedings of the 2001 Congress on Evolutionary Computation. 2001.
[18] Hu, Y., X.Yang S, . A Knowledge Based Genetic Algorithm for Path Planning of Mobile Robot. in 2004 IEEE Int. Conf. on Robotics and Automation. 2004.
[19] Gengqian, Tiejun L, P. Yuqing L, .Xiangdan P, The ant algorithm for solving robot path planning problem. in 3rd Int. Conf. on Information Technology and Applications (ICITA). 2005.
[20] 20. Ramakrishnan, R., Zein Sabatto S, , Multiple Path Planning for a Group of Mobile Robot in a 2D Environment using Genetic Algorithms. IEEE Transactions of Robotics and Systems, 2001: p. 65-71.
[21] Tu, J., X.Yang.S. Genetic Algorithms Based Path Planning for a Mobile Robot. in 2003 IEEE International Conference on Robotics & Automation. 2003. Taipei, Taiwan.
[22] Dorigo, M., L. M. Gambardella, Ant colony system: A Cooperative Learning Approach to the Traveling Salesman Problem(TSP). IEEE Transactions on Evolutionary Computation, 1997.
[23] Zheng, T.G., Huan H, Aaron S, Ant Colony System Algorithm for Real Time Globally Optimal Path Planing of Mobile Robots. ACTA AUTOMATICA SINICA, 2007. Vol.33: p. 279-285.
[24] Mei, H., Tian Y., Zu L, A Hybrid Ant Colony Optimization Algorithm for Path Planning of Robots in Dynamic Environment. Int. Journal of Information Technology, 2006,. 12(3): p. 78-88.

Norlida Buniyamin graduated from the University of Adelaide, Australia with a Bachelor Degree in Electrical and Electronic Engineering (Hons.). She was a Research Fellow with the Malaysian Institute of Microelectronic Systems (MIMOS) before joining University Teknologi MARA as a lecturer in 1988. She then obtained a M.Sc. in Industrial Control System from the University of Salford, U.K in 1993 and a PHD in the area of Knowledge Management for Manufacturing Enterprises in 2004 from the University of Manchester, Institute of Science and Technology (UMIST), United Kingdom. She is now an Associate Professor at UiTM and a Fellow of the Institution of Engineers, Malaysia (IEM). Her current research interest is in Industrial Automation and Robotics, Knowledge Management and Engineering Education.

Nohaidda Sariff graduated from the Tun Hussein Onn University, Malaysia with a Bachelor Degree in Electrical and Electronic Engineering (Automation and Robotics) in 2003. In 2008, she joined University Teknologi Mara as a lecturer of the Faculty of Electrical Engineering. She is currently pursuing a Masters Degree in Electrical Engineering at the University Teknologi Mara (UiTM), Malaysia in the areas of Mechatronics and Robotics. Her current research interests are Autonomous Mobile Robot, Navigation, and Path Planning Algorithms.

Wan Ahmad Jailani Wan Ngah graduated from the University Teknologi Malaysia with a Bachelor Degree in Computer Engineering. He is the programme Coordinator of Electronic Learning and Multimedia at the Centre for Instructor & Advanced Skill Training (CIAST) in Shah Alam, Malaysia. Jailani is currently pursuing a Masters Degree in Electrical Engineering at the University Teknologi Mara (UiTM), Malaysia in the areas of Mechatronics and Robotics.

Zainuddin Mohamad graduated with a Bachelor Degree in Mechanical Engineering in 1984 from Northrop University in the United States. Subsequently, he obtained a Degree in Aerospace Engineering followed by an MSc. in Operational Research from University of London in 1994. He is currently a lecturer of the Faculty of Mechanical Engineering, University Teknologi MARA. He has previously worked as an engineer in the Public Works Department in Malaysia as well as a development and maintenance officer in the development and maintenance department of University Teknologi Mara. Zainuddin is a member of the Institution of Engineers Malaysia and his current research interest is in Robotics, and Engineering Education.